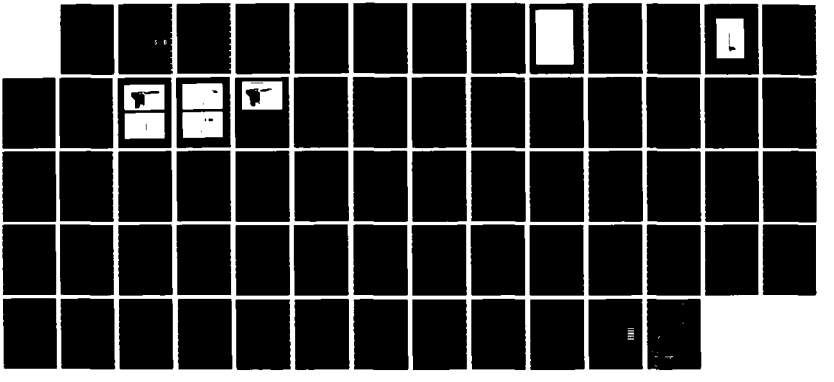
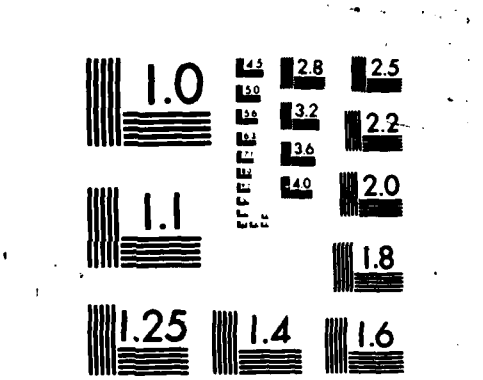
AD-A172 129 A COLOR/GRADIENT PRE- AND POST- PROCESSOR FOR 2-D FINITE ELEMENT STRESS ANALYSIS PROGRAMS(U) ARMY BALLISTIC RESEARCH LAB ABERDEEN PROVING GROUND MD

1/1

UNCLASSIFIED J M BENDER JUN 86 BRL-TR-2736 F/G 9/2 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6

US ARMY MATERIEL COMMAND

AD

(12)

TECHNICAL REPORT BRL-TR-2736

Original contains color plates: All DTIC reproductions will be in black and white

A COLOR/GRADIENT PRE- AND POST-PROCESSOR FOR 2-D FINITE ELEMENT STRESS ANALYSIS PROGRAMS

James M. Bender

June 1986

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

US ARMY BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: Technical Report BRL-TR-2736	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): A COLOR/GRADIENT PRE- AND POST-PROCESSOR FOR 2-D FINITE ELEMENT STRESS ANALYSIS PROGRAMS	5. TYPE OF REPORT & PERIOD COVERED: FINAL REPORT
6. PERFORMING ORG. REPORT NUMBER	
7. AUTHOR(s): JAMES M. BENDER	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: U.S. Army Ballistic Research Laboratory, ATTN: SLCBR-IB, Aberdeen Proving Ground, MD 21005-5066	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 1L162618AH80, 1L161102AH43
11. CONTROLLING OFFICE NAME AND ADDRESS: U.S. Army Ballistic Research Laboratory, ATTN: SLCBR-DD-T, Aberdeen Proving Ground, MD 21005-5066	12. REPORT DATE: June 1986
	13. NUMBER OF PAGES: 68
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Finite Element
Artillery Projectile
Pre-processor
Post-processor
Color Code
Stress Analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) jmk

A pre- and post-processor of finite element data for use on a color graphics microcomputer has recently been developed, within the Mechanics & Structures Branch at the U.S. Army's Ballistic Research Laboratory. The entire data file, which includes information for gridding, material properties and loading schemes, can be generated and debugged locally on a microcomputer before being sent to the mainframe computer. Before the emergence of advanced, large

(continued on other side)

(continued)

20. memory microcomputers, the preprocessing was performed interactively with the mainframe computer which consumed a considerable amount of wall-clock time. Another advantage of pre-processing on microcomputers is that the mainframe is less burdened by interactive usage and is returned to the batch mode allowing it to perform more jobs in less time.

The post-processor transforms the output data into a representative color, denoting a component, and intensity of that color, denoting a normalized stress level in the element. This allows the analyst to examine the state of stress in a multi-component structure in a single picture. For effective stress the stresses are normalized with respect to the material yield strength in each component. For the three unidirectional and shear stresses the elements comprising each component are searched for the maximum value and stresses are normalized to that value. The maximum stress levels are listed in a key at the top of the display. This process reduces the need for printed output to a backup-only basis.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

I. INTRODUCTION

The use of the finite element method of stress analysis to study artillery projectiles has been prevalent for at least the past decade. The complexity of these projectiles is growing steadily and likewise is the need for highly accurate and realistic finite element methods. What is presented here is not a new finite element code but a new, highly understandable method of grid generation (the pre-processor) and stress output display (the post-processor). The pre- and post-processor PREPPY was written within the U.S. Army's Ballistic Research Laboratory for a 2-D axisymmetric finite element code.[1] The code takes full advantage of the color graphics capabilities of a microcomputer system.

II. BACKGROUND

Until the onset of high technology warfare most artillery projectiles consisted of a high explosive housed in a steel shell and capped with a fuze. Today's modern arsenal still includes high explosive types but also includes many cargo-carrying varieties in which the payload is disseminated through an expellable base at the rear of the projectile.[2] An ejection charge housed in the nose cone is initiated by one of many types of fuzes. The expanding gases push on the payload which in turn pushes on the base and shears the screw threads coupling it to the body. The cargo, which could be grenades, land mines or smoke pellets, for example, then fall to the ground, so it is obvious the many components of these projectiles must be accurately modeled in the finite element grid.

III. PRE-PROCESSOR

One can imagine that a structure made of three different materials, say, steel, plastic and aluminum can have up to ten materials in the finite element domain. These include the three real materials and as many as seven artificial materials which could be tailored to behave as screw threads, interfaces or voids between any two of the materials. A natural course of action would be color coding as in the pre-processor of PREPPY.

PREPPY is a computer program which can be used on a microcomputer to pre-process the input data file and send it on its way to the mainframe computer undisturbed. The program enables the user to generate and de-bug the grid locally which is less costly and less time consuming than the old method of interacting with the mainframe computer. The mainframe computer also benefits in that it is not burdened with interactive usage and thereby is free to work in the batch mode for which it was designed.

---

1 Jones, R.M., Crouse, J.G., "SAAS II Finite Element Stress Analysis of Axisymmetric Solids with Orthotropic, Temperature Dependent Material Properties," Aerospace Corporation, San Bernadino, CA, September 1968.

2 Bender, J.M., Burns, B.P., "Use of the Finite Element Method of Stress Analysis as an Aid in the Design of Large and Small Caliber Projectiles," Proceedings of the 2nd Annual ASME Computer Engineering Conference, 1982.

PREPPY reads the input data file in the format identical to that used in the main program. It creates a 1000 x 11 array containing 11 bits of information about each element, including the element number, material identification number (1 through 10), the x- and y- coordinates of the four corners, and the yield strength of the element material. This array is then permanently stored on disk and is available to create color coded grids as in Figure 1 or numbered element grids as in Figure 2. The color coded grid is also accessed by the post-processor for displaying output data.

Figure 3 illustrates how the manufacturer's drawings are transformed into a finite element grid. Typically only the rear portion of projectiles are analyzed since that is where body forces accumulate in the severe in-bore phase of the launch. Generally, the loads on projectiles consist of propelling charge pressure, axial acceleration and rotating band pressure as shown in Figure 4. In the case shown, the rotating bands exert an inward-acting radial pressure of about 30 kpsi. Propellant gasses exert a pressure of about 50 kpsi to the base up to the rotating band. The resulting axial in-bore acceleration is approximately 15,000 g's .

When the user is satisifed that the grid and loading scheme are in order the microcomputer can be placed in the terminal mode and transfer the file to the mainframe computer for execution.

IV. POST-PROCESSOR

The main program to which the data file is sent includes a routine to write selected output data to a file which is easily accessed by the microcomputer acting in the terminal mode. After the file is transferred to disk the microcomputer is switched back to command mode where the post-processor of PREPPY will go to work.

Up to this point post-processing was performed by hand numbering the finite element grid and searching through reams of paper, matching printed data to the grid. If one was so inclined, colored highlighters could be used to color in elements according to material and/or stress level. This procedure can be very time consuming and tedious, especially when performing parametric studies which could result in as many as twenty output files. The post-processor of PREPPY has made this method obsolete. PREPPY will automatically generate colored stress output by the following method. First, PREPPY accesses the element information array developed during pre-processing. The eleventh column of this array contains the color code number (0 through 1) for each element. As examples, 0.1 will decode to the color red, 0.2 decodes to orange, 0.5 decodes to blue and so forth. However, this number (hue) is only one of three numbers needed for full color description. The two remaining numbers (also 0 through 1) are the intensity and luminosity. The luminosity, or brightness, is left at the value 1.0 for maximum brightness. However, the intensity value will be varied according to the stress level. The element will retain its color code but range in appearance from white through pink to red to denote low through medium to high stress levels respectively in elements color coded by red. Red for example, denotes aluminum. In a like manner, the other nine distinct colors (materials) are coded. The level of intensity is determined by accessing the 1000 x 5 stress output array which is transferred to the microcomputer's disk by the mainframe computer. This array contains the axial, radial, hoop, shear

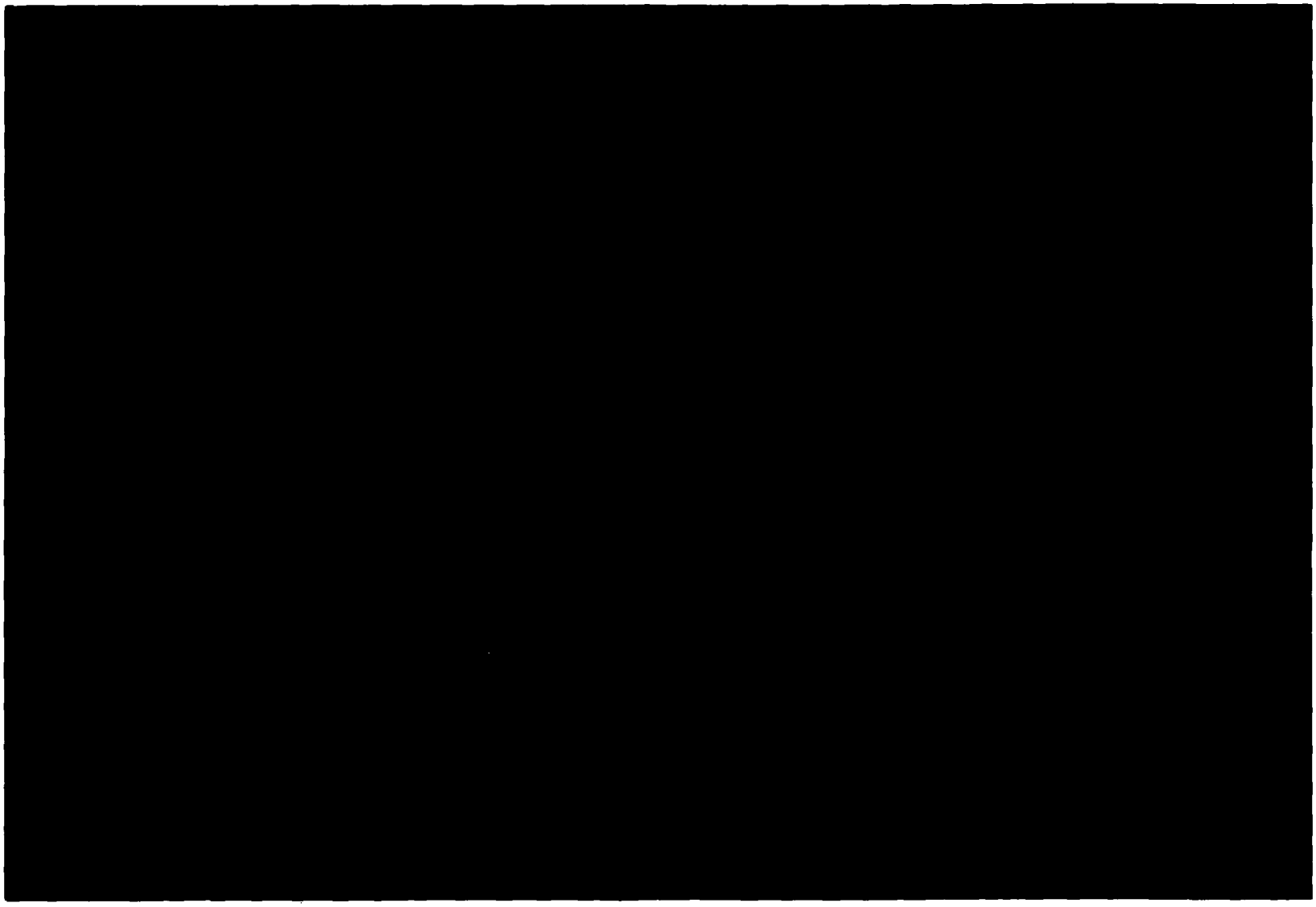


Figure 1. Color-Coded Finite Element Grid of the Rear Region of an Artillery Projectile

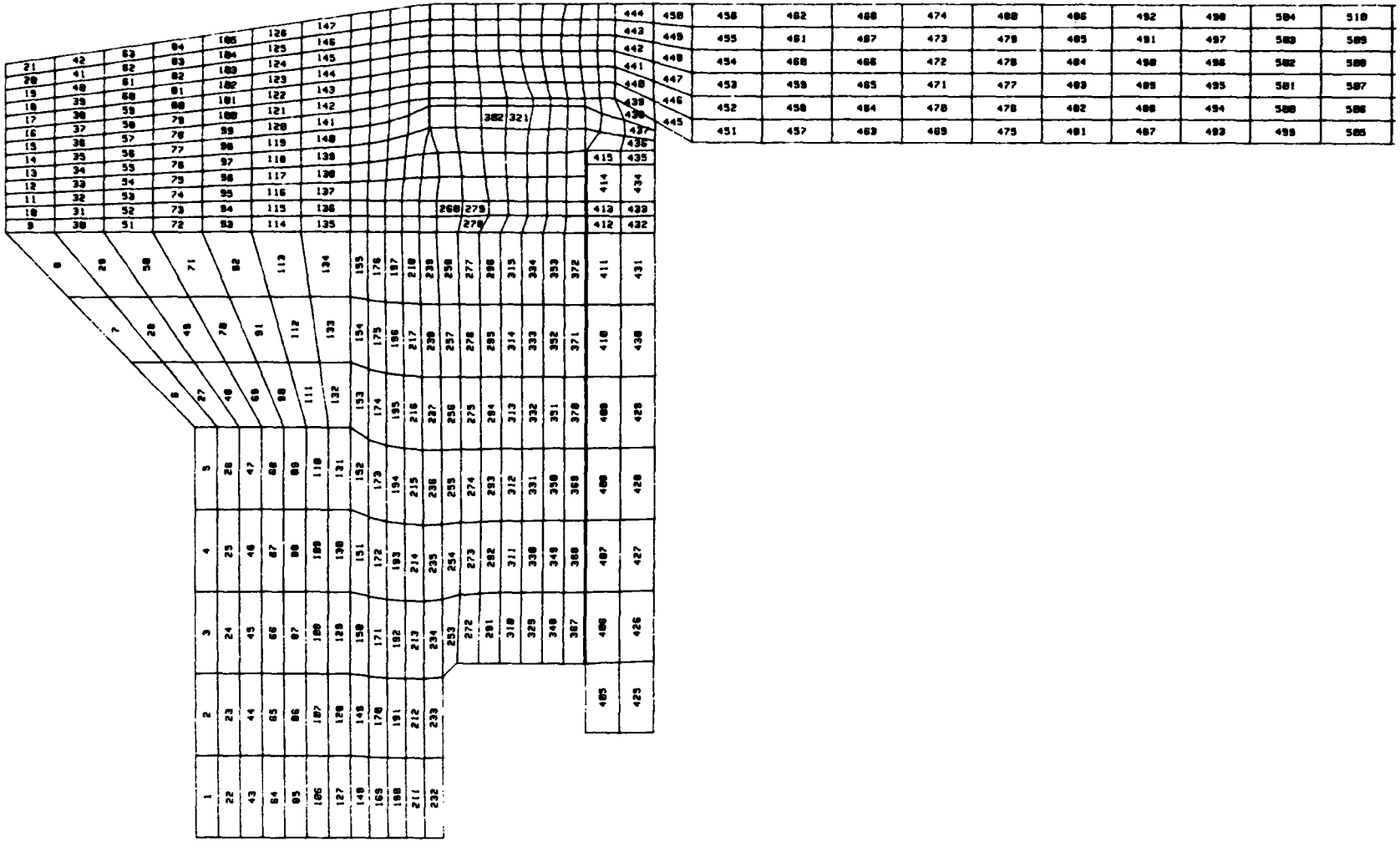


Figure 2. Numbered Element Grid

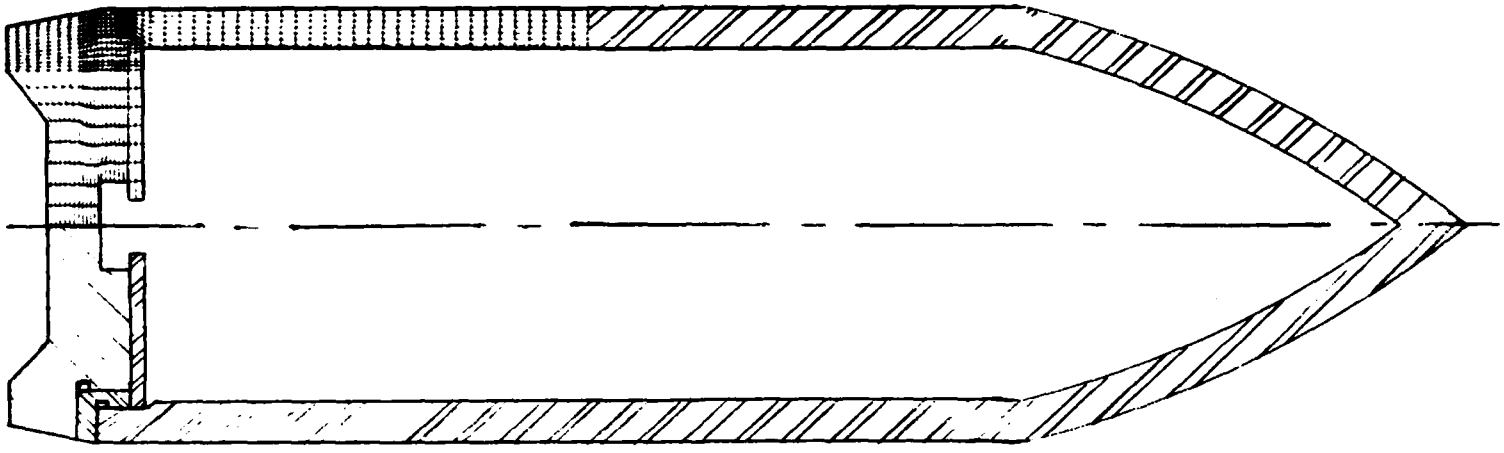


Figure 3. Drawing of Projectile and Finite Element Grid Reflection

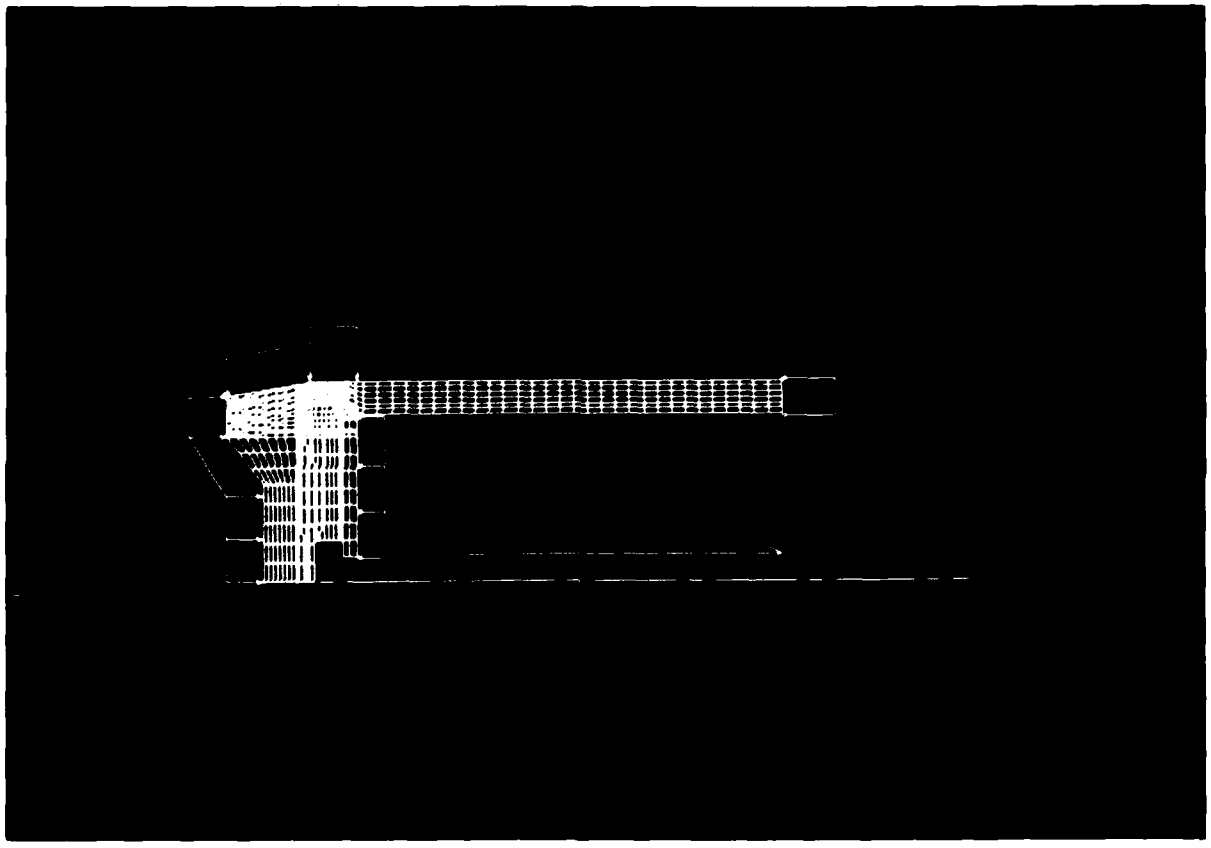


Figure 4. Loads on the Projectile During Launch

and effective stress[3,4] levels for each element. For the effective stress the value is picked off the output array and divided by the yield point of the material of the same element on the element information array. This quotient, say 0.7 for example, will create an element with 70% intensity of the designated color. Tables 1 and 2 are partial lists of the element information array and the stress output array respectively, and illustrate the procedure.

TABLE 1

Element #	Radial Stress (psi)	Hoop Stress (psi)	Axial Stress (psi)	Shear Stress (psi)	Effective Stress (psi)
345	36,264	16,035	107,892	16,514	121,031
346	38,392	14,901	106,041	16,617	107,389
347	40,392	138,771	101,581	17,722	64,993

TABLE 2

Element #	I.D. #	Coordinates LL	UL	UR	LR	Yield Stress (psi)
345	2	x,y	x,y	x,y	x,y	140,000
346	2	x,y	x,y	x,y	x,y	140,000
347	4	x,y	x,y	x,y	x,y	65,000

Referring to Tables 1 and 2, for element 346 the intensity would be 107,389/140,000 or 0.767 which decodes to orange at about 77% color intensity. Element 347 decodes to blue at 64,993/65,000 or nearly 100% color intensity. This element's stress is near the yield point. Should the value exceed 102%, an "x" is placed on the element, indicating that perhaps an elastic-plastic computation is warranted.

For stress in a uniaxial direction (axial, radial, hoop, or shear) the elements must be grouped according to identification number. Each group is searched for the maximum value in each stress direction. These maximum values are listed in a key at the top of the stress plot. The same procedure is used as in effective stress to compute intensity except that the group maximum

---

3 Hill, R., The Mathematical Theory of Plasticity, Oxford University Press, London, England, 1950.

4 Thomas, T.Y., Plastic Flow and Fracture in Solids, Academic Press, New York, NY, 1961.

value replaces the yield stress in the computation. Tables 3 and 4 illustrate the procedure where Table 3 contains the group of elements with identification number 6 and Table 4 contains pertinent portions of the element information array.

TABLE 3.

Element #	Radial Stress (psi)	Hoop Stress (psi)	Axial Stress (psi)	Shear Stress (psi)	Effective Stress (psi)
23	21,962	16,123	60,007	1692	63,222
24	23,091	16,242	59,066	2023	63,359
25	27,211	16,397	57,243	2394	63,494
196	29,997	17,291	54,129	2711	64,093
197	31,035	18,121	52,449	2638	64,125
198	33,323	19,892	50,030	2469	64,210

TABLE 4

Element #	I.D. #	Coordinates LL	UL	UR	LR	Yield Stress (psi)
23	6	x,y	x,y	x,y	x,y	65,000
24	6	x,y	x,y	x,y	x,y	65,000
25	6	x,y	x,y	x,y	x,y	65,000
196	6	x,y	x,y	x,y	x,y	65,000
197	6	x,y	x,y	x,y	x,y	65,000

In Table 3, the maximum values are boxed in. Element 23 would be green at 21,962/33,323 or 66% intensity for the radial stress plot. If the axial stress plot were generated element 23 would be green at 60,007/60,007 or 100% intensity since this is the highest stressed element in the axial direction. Table 4 is used to obtain the coordinates of the particular element.

Thus far only positive or tensile stresses have been examined. However, the MATRIX SEARCH routine includes provisions to search for minimum values (maximum compression) as well and groups these in a like manner as the tensile stresses. Just as negative (compression) values are ignored during the search for maximum tension in each group, positive values are ignored during the search for maximum compressive stresses. On the actual plots, the ignored elements are colored in at full intensity but at 20% luminosity for

completeness of the figure. Stressed elements appear much brighter. Figures 5 through 9 are examples of PREPPY's stress plots.

V. CONCLUSIONS

It can be seen that there are nine possible stress plots for each computer run: radial, hoop, axial, and shear all in either compression or tension (shear merely shows direction) and the effective stress. In most cases tensile stresses are given more attention since they tend to impact crack propagation and critical defect size.[5] In practice over the past year PREPPY has been found to help the seasoned engineer by making efficient use of time and as a visual aid in communications with the higher echelons. PREPPY has also proven itself as a valuable training tool for new engineers to help them visualize and recognize stress patterns in structures.

---

[5] Hertzberg, R.W., Deformation and Fracture Mechanics of Engineering Materials, John Wiley & Sons, NY, 1976.

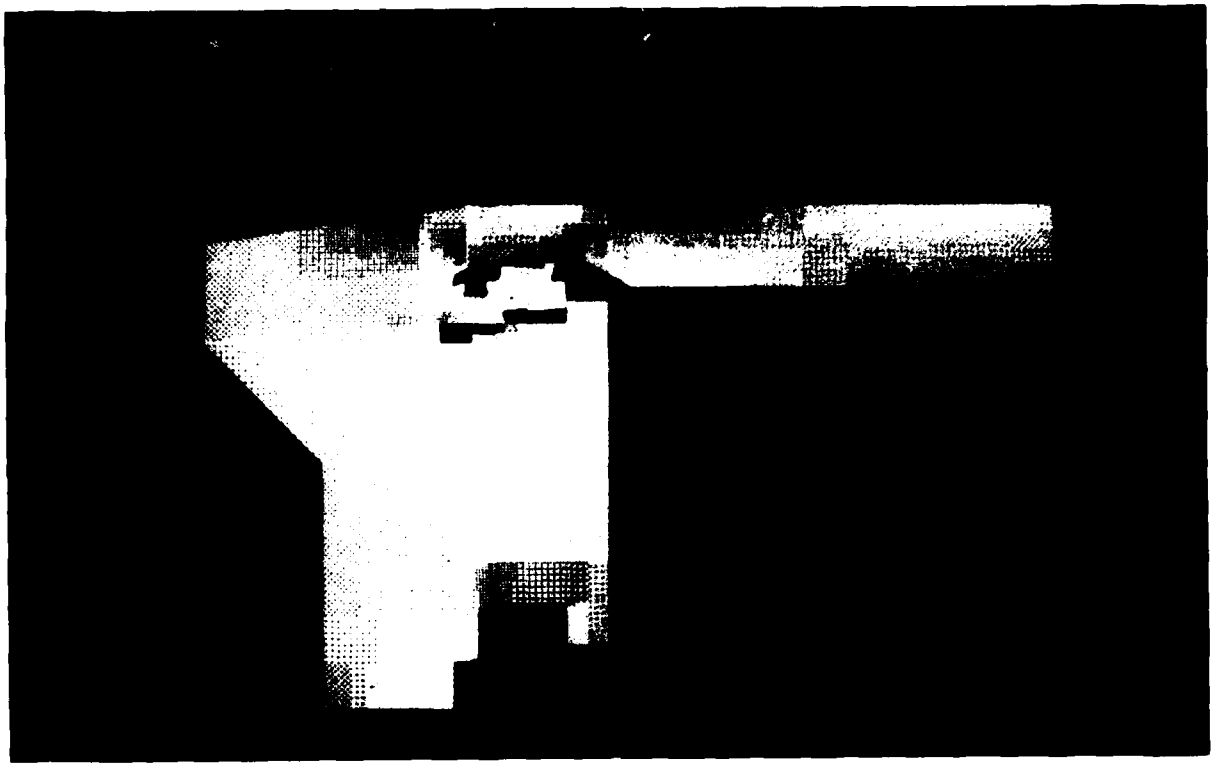


Figure 5. Compressive Axial Stresses

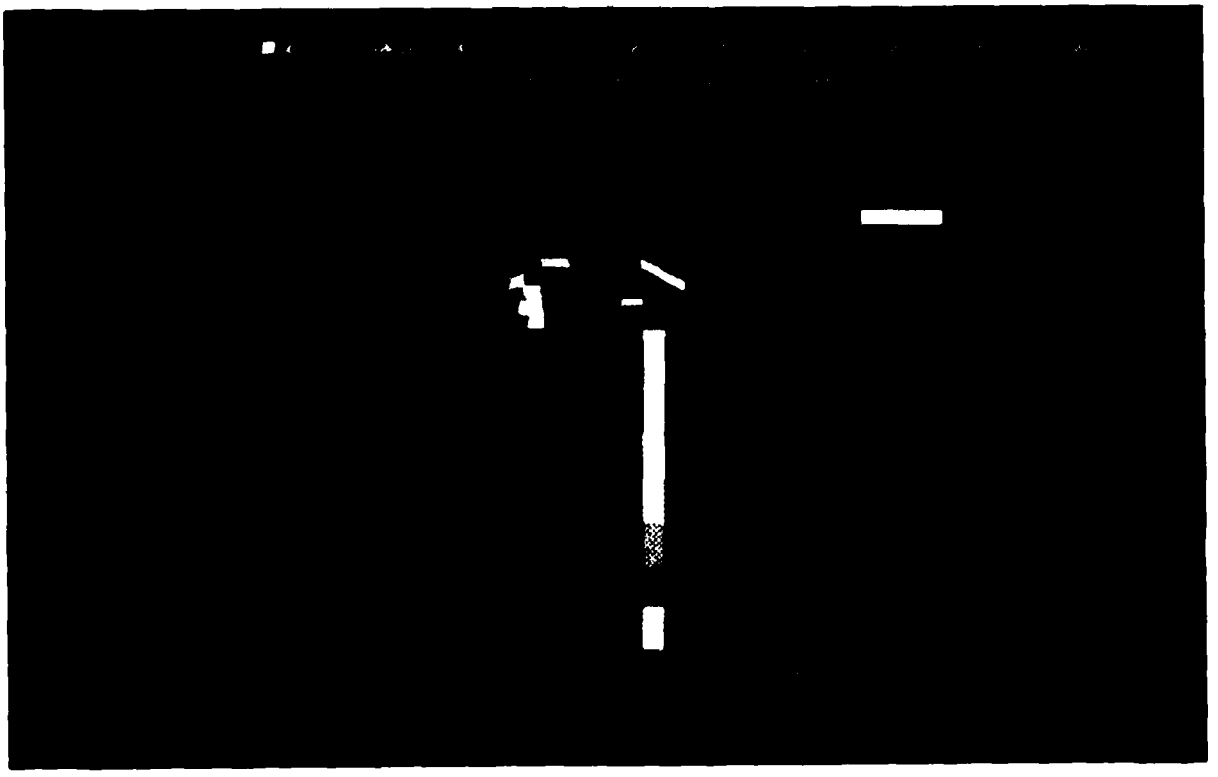


Figure 6. Tensile Radial Stresses

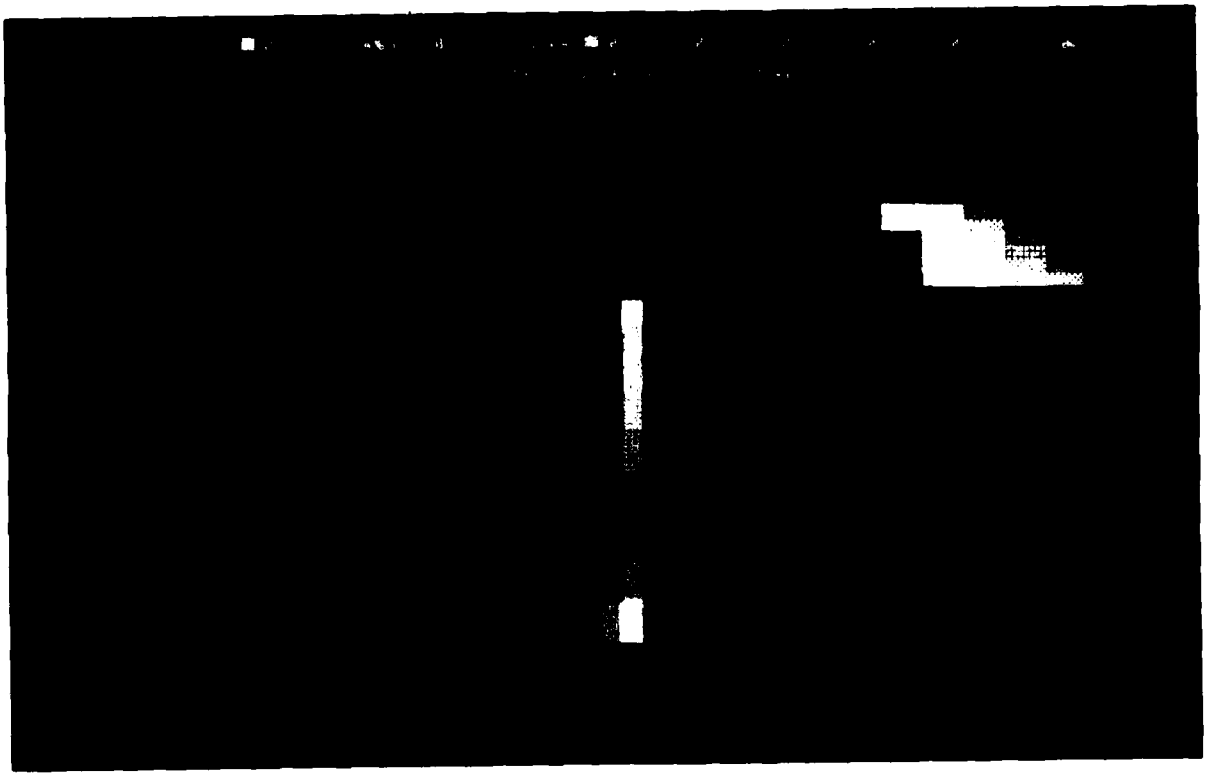


Figure 7. Tensile Hoop Stresses

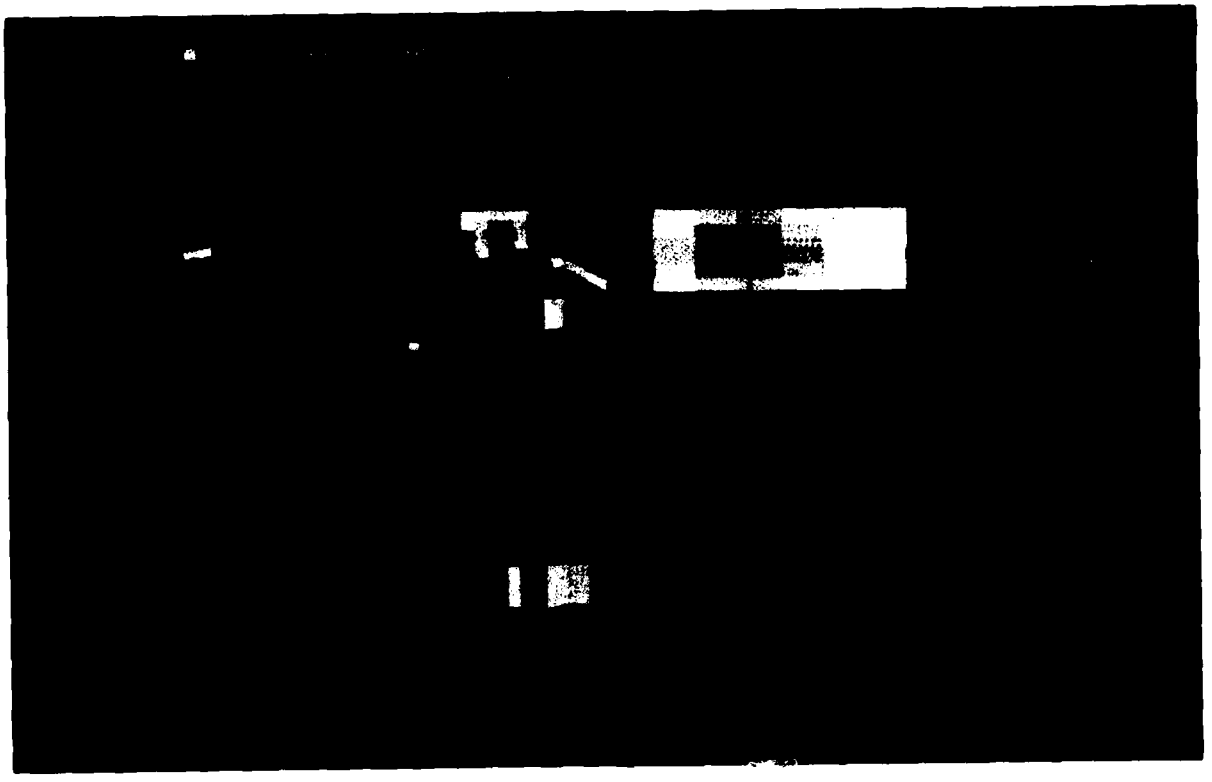


Figure 8. Positive Shear Stresses

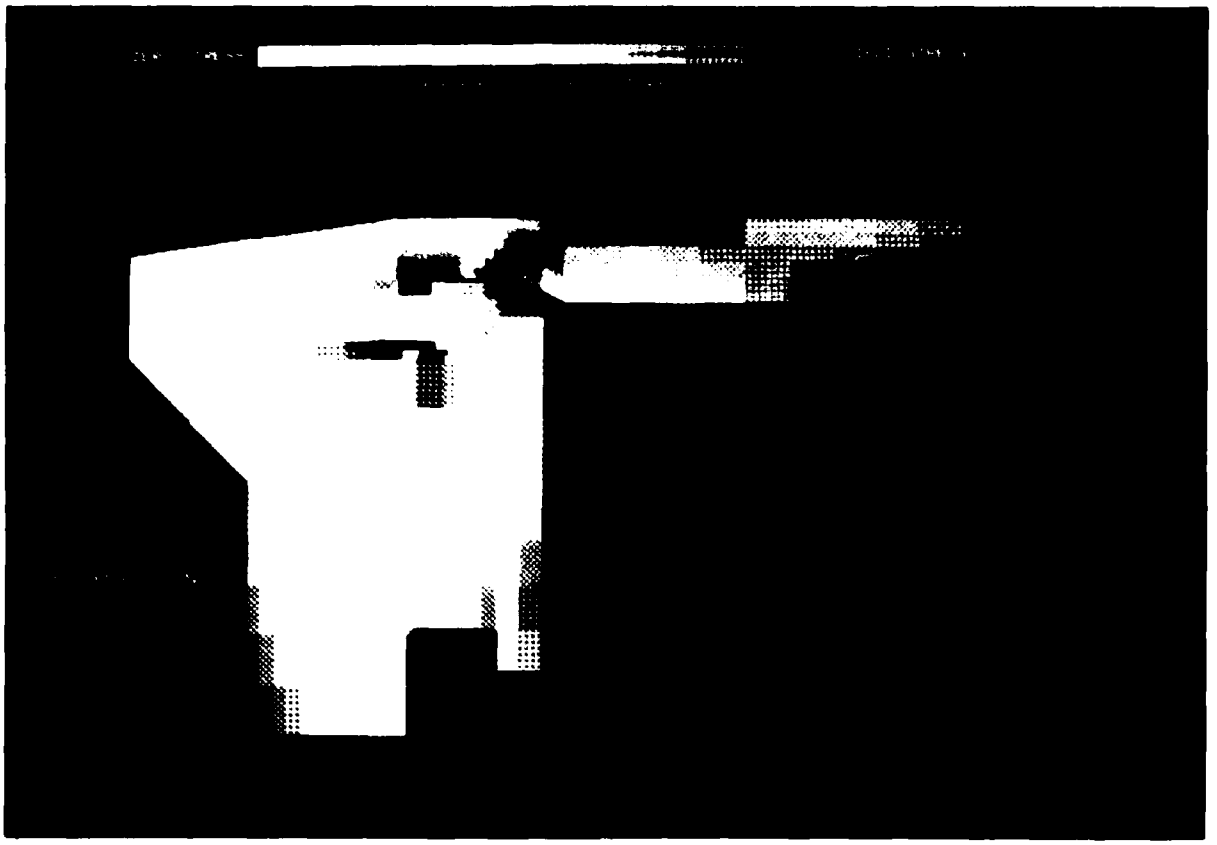


Figure 9. Effective (Von Mises) Stress

REFERENCES

1. Jones, R.M., Crouse, J.G., "SAAS II Finite Element Stress Analysis of Axisymmetric Solids with Orthotropic, Temperature Dependent Material Properties," Aerospace Corporation, San Bernadino, CA, September 1968.

2. Bender, J.M., Burns, B.P., "Use of the Finite Element Method of Stress Analysis as an Aid in the Design of Large and Small Caliber Projectiles," Proceedings of the 2nd Annual ASME Computer Engineering Conference, 1982.

3. Hill, R., The Mathematical Theory of Plasticity, Oxford University Press, London, England, 1950.

4. Thomas, T.Y., Plastic Flow and Fracture in Solids, Academic Press, New York, NY, 1961.

5. Hertzberg, R.W., Deformation and Fracture Mechanics of Engineering Materials, John Wiley & Sons, NY, 1976.

APPENDIX A

PREPPY USER'S GUIDE

1. CREATE A DATA FILE JUST AS IF YOU WERE RUNNING ON THE CDC. THIS CAN BE DONE USING RUFUS. THE TWO TYPES OF RUN STREAMS ARE LISTED IN THE APPENDIX. EACH TYPE HAS A SPECIFIC FUNCTION WHICH WILL BE DISCUSSED LATER. DO NOT INCLUDE THE END-OF-RECORD MARKS.

2. AFTER THE DATA FILE IS CREATED DO A LOAD"PREPPY:D". THE FIRST TIME THROUGH YOUR MENU SELECTION WILL BE "D" FOR DATA FILE. THIS ROUTINE WILL COMPUTE AND GENERATE YOUR GRID. IF YOU ARE HAPPY WITH THE GRID YOU CAN STORE IT AS A "GRID" FILE. ACTUALLY, YOU ARE PLOTTING INTO AN ARRAY WHICH WHEN RECALLED TAKES A FRACTION OF THE TIME TO REPLOT COMPARED TO GENERATING THE GRID. YOUR GRID WILL APPEAR ON THE CRT, HOWEVER YOU ARE GIVEN THE OPTION OF A HARD COPY. THE HARD COPY WILL BE A LARGE PLOT WHICH FITS ON USED COMPUTER PAPER OR LARGE HP PAPER RATHER WELL. PLEASE BEGIN THE NAME OF THE GRID FILE WITH A "G" SUCH AS "RING" BECOMES "GRING" AS A GRID FILE.

3. AFTER EXERCISING YOUR GRID STORAGE OPTION, PREPPY WILL CONTINUE AND BEGIN TO FILL IN THE ELEMENTS WITH A COLOR ACCORDING TO YOUR MATERIAL BLOCK ASSIGNMENT CARDS. KEEP IN MIND THAT GREEN (2), BLUE (4), AND RED (6) PHOTOGRAPH THE BEST. AS AN ATTEMPT TO STANDARDIZE THE COLOR CODING SCHEME, TRY TO RESERVE BLUE, FOR STEEL AND RED, FOR ALUMINUM. YELLOW (1) DOES NOT SHOW UP WELL SO USE IT FOR AIR. SAAS DOES NOT CARE WHAT ORDER THE MATERIAL BLOCK ASSIGNMENT CARDS OR THE MATERIAL INFORMATION CARDS ARE IN. HOWEVER, UNLIKE RUNNING SAAS ON THE CDC PREPPY IS PARTICULAR ON ASSIGNING MATERIAL TO NON-EXSISTENT ELEMENTS, SO COLOR IN ONLY THE ELEMENTS YOU HAVE. GRANTED, THIS CREATES A FEW MORE LINES, BUT THE OVERALL TIME SAVINGS IS SIGNIFICANT AND MAKES NO DIFFERENCE TO SAAS ON THE CDC.

4. IF YOU ARE HAPPY WITH THE COLOR CODED GRID, PREPPY WILL ASK YOU IF YOU WANT TO STORE A SUPERGRID. THE SUPER GRID FILE CONTAINS ALL THE INFORMATION NECESSARY FOR RECREATING THE COLOR-CODED GRID QUICKLY, THE NUMBERED ELEMENT GRID, AND THE STRESS PLOTS. IT IS SUGGESTED THAT THE NAME OF THE SUPERGRID FILE START WITH "S" SO THAT IT IS EASILY RECOGNIZED AS SUCH (EXAMPLE: THE SUPERGRID FILE OF "GAGE" BECOMES "SGAGE"). KEEP IN MIND THAT THE 9845 WILL SPEND ABOUT FOUR MINUTES COMPUTING COORDINATES FOR 300 ELEMENT GRID.

5. TO OBTAIN THE GRID FILE SELECT "G" FROM THE PREPPY MENU. ENTER THE NAME OF THE GRID FILE. SELECT THE SCALE: "FULL" MEANS THE ENTIRE FIGURE, "1/2" MEANS THE FIRST HALF, "2/4" MEANS THE SECOND QUARTER ETC, ETC. IF YOU WANT A HARD COPY SELECT THE SIZE OF THE PAPER AND THE SCALE. IF NOT, THE GRID WILL APPEAR ON THE SCREEN.

6. IF YOU WANT A NUMBERED-ELEMENT GRID, YOU MUST HAVE FIRST CREATED A SUPERGRID FILE. SIMPLY ENTER THE NAME OF THE SUPERGRID FILE, THEN SELECT THE SCALE. PLACE AN OVERSIZE SHEET OF PAPER (COMPUTER) ON THE PLATEN. THIS PROGRAM WILL NOT RUN UNLESS THE PLOTTER IS TURNED ON. BECAUSE OF JAGGIES THE NUMERALS DO NOT SHOW UP WELL ON THE CRT.

7. IF YOU ARE NOW READY FOR STRESS PLOTS, YOU MUST HAVE FIRST USED THE LONG RUNSTREAM (APPENDIX). THE THIRD ENTRY ON LINE 14 PROVIDES FOR THE OUTPUT FILE USED BY PREPPY TO GENERATE THE STRESS PLOTS. NOW IS THE TIME TO TRANSFER THE DATA FILE TO THE CDC. DON'T FORGET TO ADD IN THE END-OF-RECORD MARKS BEFORE THE HEADER LINE AND THE END. WHEN THE PROGRAM IS EXECUTED THE OUTPUT FILE (USUALLY STARTED WITH A P) WILL AUTOMATICALLY BE CREATED. THIS FILE MUST BE TRANSFERRED OVER TO THE 9845'S DISK. SELECT "S" FROM THE PREPPY MENU AND FOLLOW THE INSTRUCTIONS. MAKE ABSOLUETLY CERTAIN THAT THE OUTPUT OR P-FILE IS

COMPATIBLE WITH THE SUPERGRID FILE. TO HELP YOU KEEP FILES STRAIGHT USE THE PREPPY LOG BOOK IN RM 107, OR CREATE YOUR OWN SYSTEM. YOU CAN CREATE UP TO NINE STRESS PLOTS FOR EACH RUN: RADIAL, TANGENTIAL OR HOOP, AXIAL SHEAR (ALL IN TENSION(+) OR COMPRESSION(-), AND THE VON MISES OR EFFECTIVE STRESS. FOR THE UNIAXIAL DIRECTIONS THE HIGH EST STRESSED ELEMENT IS THE DARKEST COLORED ONE IN THE COLOR GROUP. ALL OTHER STRESSES ARE NORMALIZED TO THAT VALUE. THE MAGNITUDE OF THAT STRESS IS LISTED IN A KEY AT THE TOP OF THE PLOT. FOR THE EFF-ECTIVE STRESS ALL STRESSES ARE NORMALIZED WITH RESPECT TO THE YIELD POINT OF THE PARTICULAR MATERIAL INVOLVED. THE DARKEST COLORED ELE-MENTS ARE THOSE WHOSE VON MISES STRESS IS VERY NEAR THE YIELD POINT OF THE MATERIAL. SHOULD AN "X" APPEAR ON THE ELEMENT ITS EFFECTIVE STRESS HAS EXCEEDED THE YIELD STRESS BY TWO PER CENT.

8. SHOULD YOU NEED A BLACK-AND-WHITE VERSION OF (7), SELECT "BW" FROM FROM THE MENU. ONCE THE PLOT IS COMPLETED YOU CAN DUMP THE SCREEN ON TO THE DOT MATRIX PRINTER BY EXECUTING A "DUMP GRAPHICS". DON'T FORGET TO LABEL THE COMPONENTS OF THE GRID SINCE YOU NO LONGER HAVE THE COLOR-CODING ADVANTAGE.

PROCEDURE FOR CDC(--->9845 TRANSFERS

1. LOAD "CYBER:D",1 (OR HIT: k3 FROM ANPACK, THEN SELECT: 2, THEN ENTER: PAX THEN MFA OR JUST MFA)
2. TURN ON BLUE BOX.
3. HIT: CONT.
4. LOG IN.
5. HIT: k13.
6. HIT: STORE
7. CHANGE "TEST:T14" TO filename:C12
8. CHANGE "SIZE=010" TO "SIZE=300"
9. HIT:STORE
10. ENTER: TRMDEF,PW=81

--- 9845 TO CDC ---

11 ENTER: NEW/LFN1
12. ENTER: TEXT
13. HIT: SHIFT k14
14. HIT: N
15. AFTER UPLOAD HIT: CONTROL T
16. HIT: CONT
17. ENTER: SAVE,LFN1=filename

--- CDC TO 9845 ---

11. HIT: SHIFT k12
12. ENTER: GET filename
13. ENTER: ASSIGN,TT,B
14. TYPE: COPY,filename,B
15. HIT: SHIFT k15
16. HIT: CONT
17. AFTER FILE IS READ IN HIT: SHIFT k15

NOTE: THE ARROW KEYS ONLY APPEAR TO TO WORK WHEN TALKING WITH THE CDC. YOU MUST USE THE SPACEBAR OR BACKSPACE.

POPULAR XEDIT COMMANDS

Pn:	PRINTS n LINES
Nn:	MOVES POINTER n LINES
Dn:	DELETES n LINES STARTING AT CURRENT POINTER POSITION
T:	MOVES POINTER TO TOP OF FILE
B:	MOVES POINTER TO BOTTOM OF FILE
L/string/:	LOCATES FIRST OCCURANCE OF string
WR:	WRITES EOR MARK ABOVE CURRENT POINTER POSITION
DR:	DELETES THE EOR MARK
C/string1/string2/n:	CHANGES string1 TO string2 n TIMES (1 IS DEFAULT)

APPENDIX

SHORT RUNSTREAM:

```
JAMES,STMFZ,T800,P5.
ACCOUNT,account.
ATTACH,SAASII,ID=KOKINAKI.
BEGIN,RUNSAAS,SAASII,PL=150000,SHORT=YES.
JIM BENDER B390 ROOM 107 X6116
 SAMPLE FINITE ELEMENT GRID
  100   -2    9   90    0   72     15768
    5    9        8.       .05
 SAMPLE FINITE ELEMENT GRID
   22   54   71    1   18
  1  1   0.000   0.690  6  1   1.490   0.690                          1
  6  1   1.490   0.690  9  1   2.200   0.000                          1
  9  1   2.200   0.000 22  1   2.812   0.000                          1
 22  1   2.812   0.000 22  8   2.992   1.253                          1
 22  8   2.992   1.253 22 12   3.033   1.538                          1
```

LONG RUNSTREAM:

```
JAMES,STMFZ,T900,P5.
ACCOUNT,account.
NDFILE(1)
COPYSP,INPUT,OUTPUT.
REWIND,INPUT.
FILE,TAPE10,RT=Z,BT=C,FL=80.
BEGIN,ATTACH,PLOTLIB.
ATTACH,LGO,SAASIISHORTLGO,ID=KATHY,PW=BRUCE,MR=1.
LGO,PL=100000.
EXIT,U.
REWIND,TAPE9.
REWIND,TAPE10.
COPY,TAPE9,OUTPUT.
SAVEPF,TAPE10,Pfile,CH=PD902B,PN=PD,TY=REPLACE,UN=name,UP=passwd,ST=MFA.
BEGIN,PLOT,CALCOMP,TAPE3.
EXIT.
BEGIN,PLOT,CALCOMP,TAPE3.
JIM BENDER  B390  RM107  X6116
  SAMPLE FINITE ELEMENT GRID
  100   -2    7   37    0  72.        32.        32.    0
    5    9       20.       0.10
  SAMPLE FINITE ELEMENT GRID
   24   55   48    2   14
  1  1    .000   0.500 24  1   1.062   1.350          .000   3.426    4
 24  1   1.062   1.350 24 11   1.062   1.900                          1
 24 11   1.062   1.900 24 15   1.062   2.650                          1
 24 15   1.062   2.650 24 30   1.062   5.860                          1
 24 34   1.062   6.610 24 48   1.062  10.350                          1
```

APPENDIX B

PROGRAM LISTING

```
10    PRINTER IS 16
20    OPTION BASE 1
30  !  ..................................................................
31  !  :                                                                :
32  !  :                                                                :
40  !  :    PROGRAM: PREPPY, WRITTEN BY JAMES M. BENDER MSB, IBD, BRL ON :
50  !  :    NOVEMBER 6, 1981. THIS SECTION OF THE PROGRAM GENERATES THE  :
60  !  :    GRID FROM THE RAW DATA FILE AND CREATES THE GRID FILE AND THE :
70  !  :    SUPERGRID FILE FOR LATER USE. USER'S MANUALS ARE AVAILABLE FOR :
80  !  :    USE OF THE PROGRAM ON THE hp9845 COLOR MICRO COMPUTERS.       :
81  !  :                                                                 :
82  !  :                                                                 :
90  !  :                          REVISION DATES                         :
100 !  :                                                                 :
110 !  :   JUL 19, 1982    MAR 19, 1984                                 :
120 !  :   SEP  7, 1982    MAY  6, 1984                                 :
130 !  :   FEB  1, 1983    MAY 30, 1984                                 :
140 !  :   MAY 16, 1983    JUL 31, 1984                                 :
150 !  :   AUG 13, 1983    MAR 25, 1985                                 :
160 !  :   OCT 25, 1983    AUG 29, 1985                                 :
170 !  :   JAN 30, 1984                                                 :
171 !  :                                                                 :
180 !  ..................................................................
190   INTEGER Ncode,Material,Imin,Imax,Jmin,Jmax,Code
200   COM Ar(26,101),Az(26,101),Ncode(26,101),Elem(900,11),Material(26,101)
210   COM Nmel(100),Nemax(100),Nemin(100),Npmin(100),Npmax(100)
220   COM Numtc,Elemax,Yield(10)
230   COM Imin(100),Imax(100),Jmin(26),Jmax(26),Maxi,Maxj,Nmtl,Nbc
240   COM Numnp,Numel,Nummat
250   COM R(1000),Code(1000),Xr(1000),Z(1000),Xz(1000)
260   COM Kp,Rp(250),Zp(250)
270   COM Nele(26)
280   COM Ilow,Ihigh,Jlow,Jhigh
290   COM Npp,Nseg
300   COM File$[15],Line$[81],Que$,P$
310   DIM Line$(200)[81]
320   PRINT PAGE
330   PRINT "     Welcome to PREPPY. Please select from the menu below."
340   PRINT
350   PRINT "Choice            Meaning              Description"
360   PRINT
370   PRINT " 'D'              Data File          File from which basic grid"
380   PRINT "                                     is derived."
390   PRINT
400   PRINT " 'G'              Grid File          Stored picture of the wire"
410   PRINT "                                     frame grid."
420   PRINT
430   PRINT " 'S'            Supergrid File       Contains the information"
440   PRINT "                                     necessary for plotting color-"
450   PRINT "                                     coded grids and stress plots."
460   PRINT
470   PRINT " 'BW'           Black and White      Enables stress plots to be"
480   PRINT "                     Plots           plotted in black and white on"
490   PRINT "                                     the dot matrix printer"
500   PRINT
510   PRINT " 'EN'           Element Numbers      Sequentially numbers elements"
520   PRINT "                                     on large paper on the plotter"
530   INPUT "Enter Selection",Type$
540   PRINT PAGE
```

```
550   IF Type$="G" THEN LOAD "GRID:D",1
560   IF Type$="S" THEN LOAD "PREP1:D",1
570   IF Type$="BW" THEN LOAD "PREP2:D",1
580   IF Type$="EN" THEN LOAD "ELNUM:D",1
590   INPUT "Which Data file would you like to access?",File$
600   PRINT "You have accessed ";File$
610   File$=File$&":C12"
620   ASSIGN #1 TO File$
630   ! ************************************************************************
640   ! *                    MESH CONTROL INFORMATION                          *
650   ! ************************************************************************
660   READ #1;Line$
670   READ #1;Line$
680   READ #1;Line$
690   Run=16
700   IF Line$[1,3]="ATT" THEN Run=3
710   FOR I=1 TO Run
720   READ #1;Line$(I)
730   NEXT I
740   READ #1;Line$
750   Numtc=VAL(Line$[6,10])
760   IF Numtc=-2 THEN Numtc=1
770   Nummat=VAL(Line$[11,15])
780   Numpc=VAL(Line$[16,20])
790   Numsc=VAL(Line$[21,25])
800   READ #1;Line$
810   READ #1;Line$
820   READ #1;Line$
830   Maxi=VAL(Line$[1,5])
840   Maxj=VAL(Line$[6,10])
850   Nseg=VAL(Line$[11,15])
860   Nbc=VAL(Line$[16,20])
870   Nmtl=VAL(Line$[21,25])
880   IF P$="Z" THEN GOTO 930
890   INPUT "Do you want an input listing?(Y/N)",Que$
900   DISP "                          * Reading in Mesh Data *"
910   IF Que$="Y" THEN PRINTER IS 0
920   IF Que$="Y" THEN PRINT "Max I =";Maxi;"     Max J =";Maxj;"     # Line Seg
ment cards =";Nseg
921   IF Que$="Y" THEN PRINT
922   IF Que$="Y" THEN PRINT "  I1  J1   R1     Z1     I2  J2   R2     Z2     I3
J3    R3      Z3    Iption"
930   ! ************************************************************************
940   ! *                         INITIALIZE                                   *
950   ! ************************************************************************
960   Kp=1
970   Iseg=-1
980   Pi=PI
990   MAT Ncode=ZER
1000  MAT Ar=ZER
1010  MAT Az=ZER
1020  MAT Jmax=ZER
1030  MAT Jmin=(Maxj)
1040  MAT Imin=(Maxi)
1050  MAT Imax=ZER
1060  MAT Material=ZER
1070  MAT Elem=ZER
1080  ! ************************************************************************
1090  !                     LINE SEGMENT CARDS
1100  ! ************************************************************************
1110  Iseg=Iseg+1
1120  IF Iseg=0 THEN 1190
1130  IF (Iption<=3) OR (Iption=6) THEN GOTO 1190
1140  Rp(Kp)=R3
1150  Zp(Kp)=Z3
1160  GOTO 1190
```

```
1170  Rp(Kp)=R2
1180  Zp(Kp)=Z2
1190  IF Iseg=Nseg THEN GOTO Interior
1200  Ic=3
1210  I1=I2=I3=J1=J2=J3=R1=R2=R3=Z1=Z2=Z3=0
1220  READ #1;Line$
1230  I1=VAL(Line$[1,3])
1240  J1=VAL(Line$[4,6])
1250  R1=VAL(Line$[7,14])
1260  Z1=VAL(Line$[15,22])
1270  Iption=VAL(Line$[67,71])
1280  IF Iption=0 THEN Next_ic
1290  I2=VAL(Line$[23,25])
1300  J2=VAL(Line$[26,28])
1310  R2=VAL(Line$[29,36])
1320  Z2=VAL(Line$[37,44])
1330  IF (Iption<>4) AND (Iption<>3) AND (Iption<>13) THEN Next_ic
1340  R3=VAL(Line$[51,58])
1350  Z3=VAL(Line$[59,66])
1360  IF Iption=4 THEN Next_ic
1370  I3=VAL(Line$[45,47])
1380  J3=VAL(Line$[48,50])
1390 Next_ic: Ic=Ic+1
1400  IF (Que$="N") OR (P$="Z") THEN GOTO 1430
1410  PRINT USING Image;I1,J1,R1,Z1,I2,J2,R2,Z2,I3,J3,R3,Z3,Iption
1420 Image:IMAGE 1X,DDD,1X,DDD,1X,DD.DDD,1X,DD.DDD,1X,DDD,1X,DDD,1X,DD.DDD,1X,DD
.DDD,1X,DDD,1X,DDD,1X,DDD.DDD,1X,DDD.DDD,3X,DD
1430  IF Iption=0 THEN GOTO Diagonal
1440  IF Iption=11 THEN GOTO 1680
1450  IF (Iption=5) OR (Iption=1) THEN Straight
1460  IF Iption=2 THEN GOTO Diagonal
1470  IF Iption=13 THEN GOTO Diagonal
1480  Rp(Kp)=R1
1490  Zp(Kp)=Z1
1500  GOTO Diagonal
1510 !
1520 ! ***** POINTS *****
1530 !
1540  Rp(Kp)=R1
1550  Zp(Kp)=Z1
1560  Kp=Kp+1
1570  IF Iseg+1<Nseg THEN GOTO Diagonal
1580  Rp(Kp)=Rp(1)
1590  Zp(Kp)=Zp(1)
1600  GOTO Diagonal
1610 Straight:    ! ***** FOR IPTION = 1 *****
1620 !
1630  Rp(Kp)=R1
1640  Zp(Kp)=Z1
1650  Rp(Kp+1)=R2
1660  Zp(Kp+1)=Z2
1670  Kp=Kp+2
1680  IF Iption=11 THEN Iption=1
1690 !
1700 Diagonal:     ! ***** FOR IPTION = 0,2 or 13 *****
1710 !
1720  IF I1=-1 THEN GOTO Interior
1730  IF Iption=13 THEN Ipt=13
1740  IF Iption=13 THEN Iption=3
1750  Iption=Iption+1
1760  Ar(I1,J1)=R1
1770  Az(I1,J1)=Z1
1780  Ncode(I1,J1)=1
1790  CALL Mnimx(I1,J1)
1800  ON Iption GOTO 1110,1840,1840,Circle,Circle,1840
1810 ! ************************************************************************
```

```
1820 !                    GENERATE STRAIGHT LINES ON BOUNDARY
1830 ! ************************************************************************
1840  Di=ABS(I2-I1)
1850  Dj=ABS(J2-J1)
1860  Ar(I2,J2)=R2
1870  Az(I2,J2)=Z2
1880  Ncode(I2,J2)=1
1890  CALL Mnimx(I2,J2)
1900  Istrt=I1
1910  Istp=I2
1920  Jstrt=J1
1930  Jstp=J2
1940  Diff=MAX(Di,Dj)
1950  Iter=Diff-1
1960  Iinc=0
1970  Jinc=0
1980  IF I2<>I1 THEN Iinc=(I2-I1)/ABS(I2-I1)
1990  IF J2<>J1 THEN Jinc=(J2-J1)/ABS(J2-J1)
2000  Kappa=1
2010  IF (I2<>I1) AND (J2<>J1) AND (Iption<>3) THEN Kappa=2
2020  IF Kappa=2 THEN Diff=Diff*2
2030  Rinc=(R2-R1)/Diff
2040  Zinc=(Z2-Z1)/Diff
2050 !
2060 !           ***** CHECK FOR INPUT ERROR *****
2070 !
2080  IF (Kappa<>2) OR (Di=Dj) THEN GOTO 2130
2090  GOTO 1110
2100 !
2110 !                ***** INTERPOLATE *****
2120 !
2130  I=I1
2140  J=J1
2150  FOR M=1 TO Iter
2160  IF (Iter=0) AND (Iption=2) THEN GOTO 2550
2170  IF (Iter=0) AND (Iption=6) THEN GOTO 2550
2180  IF Kappa=2 THEN GOTO 2280
2190  Iold=I
2200  I=I+Iinc
2210  Jold=J
2220  J=J+Jinc
2230  Ar(I,J)=Ar(Iold,Jold)+Rinc
2240  Az(I,J)=Az(Iold,Jold)+Zinc
2250  CALL Mnimx(I,J)
2260  Ncode(I,J)=1
2270  GOTO 2550
2280  IF (I1>I2) AND (Iption=2) THEN GOTO 2430
2290  IF (I1<I2) AND (Iption=6) THEN GOTO 2430
2300  Iold=I
2310  I=I+Iinc
2320  Ar(I,J)=Ar(Iold,J)+Rinc
2330  Az(I,J)=Az(Iold,J)+Zinc
2340  Ncode(I,J)=1
2350  CALL Mnimx(I,J)
2360  Jold=J
2370  J=J+Jinc
2380  Ar(I,J)=Ar(I,Jold)+Rinc
2390  Az(I,J)=Az(I,Jold)+Zinc
2400  Ncode(I,J)=1
2410  CALL Mnimx(I,J)
2420  GOTO 2550
2430  Jold=J
2440  J=J+Jinc
2450  Ar(I,J)=Ar(I,Jold)+Rinc
2460  Az(I,J)=Az(I,Jold)+Zinc
2470  Ncode(I,J)=1
```

```
2480  CALL Mnimx(I,J)
2490  Iold=I
2500  I=I+Iinc
2510  Ar(I,J)=Ar(Iold,J)+Rinc
2520  Az(I,J)=Az(Iold,J)+Zinc
2530  Ncode(I,J)=1
2540  CALL Mnimx(I,J)
2550  NEXT M
2560  IF Kappa=1 THEN GOTO 1110
2570  IF (I1>I2) AND (Iption=2) THEN GOTO 2640
2580  IF (I1<I2) AND (Iption=6) THEN GOTO 2640
2590  Iold=I
2600  I=I+Iinc
2610  Ar(I,J)=Ar(Iold,J)+Rinc
2620  Az(I,J)=Az(Iold,J)+Zinc
2630  GOTO 2680
2640  Jold=J
2650  J=J+Jinc
2660  Ar(I,J)=Ar(I,Jold)+Rinc
2670  Az(I,J)=Az(I,Jold)+Zinc
2680  Ncode(I,J)=1
2690  CALL Mnimx(I,J)
2700  GOTO 1110
2710 ! ************************************************************************
2720 ! *                    GENERATE CIRCULAR ARCS ON BOUNDARY                *
2730 ! ************************************************************************
2740 Circle: !
2750  Ar(I2,J2)=R2
2760  Az(I2,J2)=Z2
2770  Ncode(I2,J2)=1
2780  CALL Mnimx(I2,J2)
2790  IF Iption=5 THEN GOTO 3060
2800 !
2810 !                     ***** FIND CENTER OF CIRCLE *****
2820 !
2830  Ar(I3,J3)=R3
2840  Az(I3,J3)=Z3
2850  Ncode(I3,J3)=1
2860  CALL Mnimx(I3,J3)
2870  Slac=(Z2-Z1)/(R2-R1)
2880  Slbf=-1/Slac
2890  Slce=(Z3-Z2)/(R3-R2)
2900  Sldf=-1/Slce
2910 !
2920 !                    ***** CHECK FOR INPUT ERROR *****
2930 !
2940  IF ABS(Slac-Slce)>.001 THEN GOTO 2960
2950  GOTO 1110
2960  R4=R1+(R2-R1)/2
2970  Z4=Z1+(Z2-Z1)/2
2980  R5=R2+(R3-R2)/2
2990  Z5=Z2+(Z3-Z2)/2
3000  Bbf=Z4-Slbf*R4
3010  Bdf=Z5-Sldf*R5
3020  Rc=(Bbf-Bdf)/(Sldf-Slbf)
3030  Zc=Slbf*Rc+Bbf
3040  Kappa=1
3050  GOTO 3090
3060  Kappa=2
3070  Rc=R3
3080  Zc=Z3
3090  Istrt=I1
3100  Istp=I2
3110  Jstrt=J1
3120  Jstp=J2
3130  Rstrt=R1
```

```
3140  Rstp=R2
3150  Zstrt=Z1
3160  Zstp=Z2
3170  CALL Angle(Rstrt,Zstrt,Rc,Zc,Ang1)
3180  CALL Angle(Rstp,Zstp,Rc,Zc,Ang2)
3190  IF (ABS(Ang2)<=.00001) AND (Ang1>PI) THEN Ang2=2*PI
3200  IF (ABS(Ang1)<=.00001) AND (Ang2>PI) THEN Ang1=2*PI
3210 !
3220 !              ***** FIND ANGULAR INCREMENT *****
3230 !
3240  Di=ABS(Istp-Istrt)
3250  Dj=ABS(Jstp-Jstrt)
3260  Iinc=0
3270  Jinc=0
3280  IF Istrt<>Istp THEN Iinc=(Istp-Istrt)/ABS(Istp-Istrt)
3290  IF Jstrt<>Jstp THEN Jinc=(Jstp-Jstrt)/ABS(Jstp-Jstrt)
3300  Lamda=1
3310  IF (Iinc<>0) AND (Jinc<>0) THEN Lamda=2
3320  Diff=MAX(Di,Dj)
3330  Iter=Diff-1
3340  IF Lamda=2 THEN Diff=2*Diff
3350  IF Ang2-Ang1>PI THEN Ang2=Ang2-2*PI
3360  IF Ang2-Ang1<-PI THEN Ang2=Ang2+2*PI
3370  Delphi=(Ang2-Ang1)/Diff
3380 !
3390 !               ***** CHECK FOR INPUT ERROR *****
3400 !
3410  IF (Lamda<>2) OR (Di=Dj) THEN GOTO 3430
3420  GOTO 1110
3430  Io=Istrt
3440  Jo=Jstrt
3450 !
3460 !                    ***** INTERPOLATE *****
3470 !
3480  FOR M=1 TO Iter
3490  IF Lamda=2 THEN GOTO 3600
3500  I=Io+Iinc
3510  J=Jo+Jinc
3520  CALL Mnimx(I,J)
3530  Ncode(I,J)=1
3540  CALL Circles(Ang1,Delphi,Rstrt,Zstrt,Rc,Zc,I,J)
3550  IF Ipt=13 THEN GOTO 3770
3560  Zp(Kp)=Az(I,J)
3570  Rp(Kp)=Ar(I,J)
3580  Kp=Kp+1
3590  GOTO 3770
3600  I=Io+Iinc
3610  J=Jo
3620  Ncode(I,J)=1
3630  CALL Mnimx(I,J)
3640  CALL Circles(Ang1,Delphi,Rstrt,Zstrt,Rc,Zc,I,J)
3650  IF Ipt=13 THEN GOTO 3690
3660  Zp(Kp)=Az(I,J)
3670  Rp(Kp)=Ar(I,J)
3680  Kp=Kp+1
3690  J=Jo+Jinc
3700  Ncode(I,J)=1
3710  CALL Mnimx(I,J)
3720  CALL Circles(Ang1,Delphi,Rstrt,Zstrt,Rc,Zc,I,J)
3730  IF Ipt=13 THEN GOTO 3770
3740  Zp(Kp)=Az(I,J)
3750  Rp(Kp)=Ar(I,J)
3760  Kp=Kp+1
3770  Io=I
3780  Jo=J
3790  NEXT M
```

```
3800  IF Lamda<>2 THEN GOTO 3890
3810  I=Io+Iinc
3820  Ncode(I,J)=1
3830  CALL Mnimx(I,J)
3840  CALL Circles(Angl,Delphi,Rstrt,Zstrt,Rc,Zc,I,J)
3850  IF Ipt=13 THEN GOTO 3890
3860  Zp(Kp)=Az(I,J)
3870  Rp(Kp)=Ar(I,J)
3880  Kp=Kp+1
3890  IF Kappa=2 THEN GOTO 1110
3900  Istrt=I2
3910  Istp=I3
3920  Jstrt=J2
3930  Jstp=J3
3940  Rstrt=R2
3950  Rstp=R3
3960  Zstrt=Z2
3970  Zstp=Z3
3980  Kappa=2
3990  IF Iption=5 THEN GOTO 4040
4000  IF Ipt=13 THEN GOTO 4040
4010  Rp(Kp)=R2
4020  Zp(Kp)=Z2
4030  Kp=Kp+1
4040  GOTO 3170
4050 ! ************************************************************************
4060 ! *          CALCULATE COORDINATES OF INTERIOR POINTS                    *
4070 ! ************************************************************************
4080 Interior:PRINTER IS 16
4090  DISP "                          * Computing Node Coordinates *"
4100  IF Maxj<=2 THEN GOTO 4300
4110  J2=Maxj-1
4120  FOR N=1 TO 500
4130  Resid=0
4140  FOR J=2 TO J2
4150  I1=Imin(J)+1
4160  I2=Imax(J)-1
4170  FOR I=I1 TO I2
4180  IF Ncode(I,J)=1 THEN GOTO 4240
4190  Dr=(Ar(I+1,J)+Ar(I-1,J)+Ar(I,J+1)+Ar(I,J-1))/4-Ar(I,J)
4200  Dz=(Az(I+1,J)+Az(I-1,J)+Az(I,J+1)+Az(I,J-1))/4-Az(I,J)
4210  Resid=Resid+ABS(Dr)+ABS(Dz)
4220  Ar(I,J)=Ar(I,J)+1.8*Dr
4230  Az(I,J)=Az(I,J)+1.8*Dz
4240  NEXT I
4250  NEXT J
4260  IF N=1 THEN Res1=Resid
4270  IF (N=1) AND (Resid=0) THEN GOTO 4300
4280  IF Resid/Res1<.00001 THEN GOTO 4300
4290  NEXT N
4300  CALL Points(#1)
4310  FOR Ki=1 TO 25
4320  FOR Kj=1 TO 100
4330  IF Ncode(Ki,Kj)=4 THEN GOTO 4350
4340  Ncode(Ki,Kj)=0
4350  NEXT Kj
4360  NEXT Ki
4370  FOR Ki=1 TO 25
4380  FOR Kj=1 TO 100
4390  Ip1=Ki+1
4400  Jp1=Kj+1
4410  Ndtest=4
4420  IF Ncode(Ki,Kj)<>4 THEN GOTO 4480
4430  IF (Ip1>26) OR (Jp1>101) THEN GOTO 4480
4440  IF Ncode(Ki,Jp1)=4 THEN Ndtest=2
4450  IF Ncode(Ip1,Kj)=4 THEN Ndtest=1
```

```
4460  IF (Ncode(Ip1,Kj)=4) AND (Ncode(Ki,Jp1)=4) THEN Ndtest=3
4470  Ncode(Ki,Kj)=Ndtest
4480  NEXT Kj
4490  NEXT Ki
4500  Ilow=1
4510  Ihigh=25
4520  Jlow=1
4530  Jhigh=100
4540  CALL Plots(#1,F$)
4550              W=1.5
4560              BEEP
4570              WAIT 300/W
4580              BEEP
4590              WAIT 150/W
4600              BEEP
4610              WAIT 150/W
4620              BEEP
4630              WAIT 300/W
4640              BEEP
4650              WAIT 600/W
4660              BEEP
4670              WAIT 300/W
4680              BEEP
4690              PEN 0
4700 END
4710        SUB Mnimx(I,J)
4720        OPTION BASE 1
4730        INTEGER Ncode,Material,Imin,Imax,Jmin,Jmax,Code
4740        COM Ar(26,101),Az(26,101),Ncode(26,101),Elem(900,11),Material(26,101)
4750        COM Nmel(100),Nemax(100),Nemin(100),Npmin(100),Npmax(100)
4760        COM Numtc,Elemax,Yield(10)
4770        COM Imin(100),Imax(100),Jmin(26),Jmax(26),Maxi,Maxj,Nmtl,Nbc
4780        COM Numnp,Numel,Nummat
4790        COM R(1000),Code(1000),Xr(1000),Z(1000),Xz(1000)
4800        COM Kp,Rp(250),Zp(250)
4810        COM Nele(26)
4820        COM Ilow,Ihigh,Jlow,Jhigh
4830        COM Npp,Nseg
4840        COM File$[15],Line$[81],Que$,P$
4850        IF J<Jmin(I) THEN Jmin(I)=J
4860        IF J>Jmax(I) THEN Jmax(I)=J
4870        IF I<Imin(J) THEN Imin(J)=I
4880        IF I>Imax(J) THEN Imax(J)=I
4890        SUBEXIT
4900        SUBEND
4910  SUB Points(#1)
4920  OPTION BASE 1
4930  INTEGER Ncode,Material,Imin,Imax,Jmin,Jmax,Code
4940  COM Ar(26,101),Az(26,101),Ncode(26,101),Elem(900,11),Material(26,101)
4950  COM Nmel(100),Nemax(100),Nemin(100),Npmin(100),Npmax(100)
4960  COM Numtc,Elemax,Yield(10)
4970  COM Imin(100),Imax(100),Jmin(26),Jmax(26),Maxi,Maxj,Nmtl,Nbc
4980  COM Numnp,Numel,Nummat
4990  COM R(1000),Code(1000),Xr(1000),Z(1000),Xz(1000)
5000  COM Kp,Rp(250),Zp(250)
5010  COM Nele(26)
5020  COM Ilow,Ihigh,Jlow,Jhigh
5030  COM Npp,Nseg
5040  COM File$[15],Line$[81],Que$,P$
5050  Zero=.0000001
5060  ! ************************************************************************
5070  ! *            ESTABLISH NODAL POINT INFORMATION                         *
5080  ! ************************************************************************
5090  Nel=0
5100  Nodsum=0
5110  FOR J=1 TO Maxj
```

```
5120  Nstart=Imin(J)
5130  Nstop=Imax(J)
5140  FOR I=Nstart TO Nstop
5150  Nodsum=Nodsum+1
5160  NEXT I
5170  NEXT J
5180  Nelsum=0
5190  Jjmax=Maxj-1
5200  FOR Jj=1 TO Jjmax
5210  Nstop=MIN(Imax(Jj),Imax(Jj+1))-1
5220  Nstart=MAX(Imin(Jj),Imin(Jj+1))
5230  FOR Ii=Nstart TO Nstop
5240  Nelsum=Nelsum+1
5250  NEXT Ii
5260  NEXT Jj
5270  Numnp=Nodsum
5280  Numel=Nelsum
5290  FOR J=1 TO Maxj
5300  Nstart=Imin(J)
5310  Nstop=Imax(J)
5320  FOR I=Nstart TO Nstop
5330  Np=FNNode(I,J)
5340  R(Np)=Ar(I,J)
5350  Z(Np)=Az(I,J)
5360  NEXT I
5370  NEXT J
5380 ! ************************************************************************
5390 ! *              READ AND ASSIGN BOUNDARY CONDITIONS                     *
5400 ! ************************************************************************
5410 ! *                          INITIALIZE
*
5420 ! ************************************************************************
5430  FOR I=1 TO Numnp
5440  Code(I)=0
5450  IF ABS((R(I)<=Zero) AND (Npp=0)) THEN Code(I)=1
5460  Xr(I)=0
5470  Xz(I)=0
5480  NEXT I
5490  IF Nbc=0 THEN GOTO 5680
5500  FOR Ibcon=1 TO Nbc
5510  READ #1;Line$
5520  I1=VAL(Line$[2,6])
5530  I2=VAL(Line$[7,11])
5540  J1=VAL(Line$[12,16])
5550  J2=VAL(Line$[17,21])
5560 ! Con=VAL(Line$[22,31])
5570 ! Rcon=VAL(Line$[32,41])
5580 ! Zcon=VAL(Line$[42,51])
5590  FOR I=I1 TO I2
5600  FOR J=J1 TO J2
5610  Np=FNNode(I,J)
5620  Code(Np)=Con
5630  Xr(Np)=Rcon
5640  Xz(Np)=Zcon
5650  NEXT J
5660  NEXT I
5670  NEXT Ibcon
5680  FOR J=1 TO Maxj
5690  Nstart=Imin(J)
5700  Nstop=Imax(J)
5710  FOR I=Nstart TO Nstop
5720  Np=FNNode(I,J)
5730  Ncode(I,J)=4
5740  Ar(I,J)=R(Np)
5750  Az(I,J)=Z(Np)
5760  NEXT I
```

```
5770  NEXT J
5780 ! Nbc=Nbc1
5790  SUBEXIT
5800  SUBEND
5810       SUB Angle(R,Z,Rc,Zc,Ang)
5820       OPTION BASE 1
5830       INTEGER Ncode,Material,Imin,Imax,Jmin,Jmax,Code
5840       COM Ar(26,101),Az(26,101),Ncode(26,101),Elem(900,11),Material(26,101)
5850       COM Nmel(100),Nemax(100),Nemin(100),Npmin(100),Npmax(100)
5860       COM Numtc,Elemax,Yield(10)
5870       COM Imin(100),Imax(100),Jmin(26),Jmax(26),Maxi,Maxj,Nmtl,Nbc
5880       COM Numnp,Numel,Nummat
5890       COM R(1000),Code(1000),Xr(1000),Z(1000),Xz(1000)
5900       COM Kp,Rp(250),Zp(250)
5910       COM Nele(26)
5920       COM Ilow,Ihigh,Jlow,Jhigh
5930       COM Npp,Nseg
5940       COM File$[15],Line$[81],Que$,P$
5950       D1=Z-Zc
5960       D2=R-Rc
5970       IF ABS(R-Rc)>.00000001 THEN GOTO 6050
5980       Ang=PI/2
5990       IF D1>.00000001 THEN SUBEXIT
6000       Ang=-Ang
6010       SUBEXIT
6020       !
6030       !          ***** ALLOW CIRCLE TO CROSS AXIS *****
6040       !
6050       Ang=ATN(D1/D2)
6060       IF (D1>=0) AND (D2<0) THEN Ang=Ang+PI
6070       IF (D1<=0) AND (D2<0) THEN Ang=Ang-PI
6080       SUBEXIT
6090       SUBEND
6100     DEF FNNode(I,J) !  *****  THIS IS THE SUBFUNCTION Node *****
6110     OPTION BASE 1
6120     INTEGER Ncode,Material,Imin,Imax,Jmin,Jmax,Code
6130     COM Ar(26,101),Az(26,101),Ncode(26,101),Elem(900,11),Material(26,101)
6140     COM Nmel(100),Nemax(100),Nemin(100),Npmin(100),Npmax(100)
6150     COM Numtc,Elemax,Yield(10)
6160     COM Imin(100),Imax(100),Jmin(26),Jmax(26),Maxi,Maxj,Nmtl,Nbc
6170     COM Numnp,Numel,Nummat
6180     COM R(1000),Code(1000),Xr(1000),Z(1000),Xz(1000)
6190     COM Kp,Rp(250),Zp(250)
6200     COM Nele(26)
6210     COM Ilow,Ihigh,Jlow,Jhigh
6220     COM Npp,Nseg
6230     COM File$[15],Line$[81],Que$,P$
6240     Node=0
6250     FOR Jj=1 TO J
6260     Nstart=Imin(Jj)
6270     Nstop=Imax(Jj)
6280     FOR Ii=Nstart TO Nstop
6290     Node=Node+1
6300     IF (Jj=J) AND (Ii=I) THEN RETURN Node
6310     NEXT Ii
6320     NEXT Jj
6330     RETURN Node
6340     FNEND
6350   SUB Plots(#1,F$)
6360   OPTION BASE 1
6370   INTEGER Ncode,Material,Imin,Imax,Jmin,Jmax,Code
6380   PLOTTER IS "GRAPHICS"
6390   GRAPHICS
6400   BEEP
6410   IF P$="Z" THEN GOTO 6530
6420   INPUT "Do you want to create a Grid File?(Y/N)",R$
```

```
6430  IF R$="N" THEN GOTO 6460
6440  DIM Grid(2000,3)
6450  PLOTTER IS Grid(*)
6460  INPUT "Hard copy(Y/N)",Ans$
6470  IF Ans$="N" THEN GOTO 6530
6480  PEN 1
6490  OUTPUT 7,5;"IP1000,0,15102,11400"
6500  PLOTTER IS "9872A"
6510  PRINT "Prepare platten for LARGE paper then  C "
6511  PRINT "                                      O "
6512  PRINT "                                      N "
6513  PRINT "                                      T "
6520  PAUSE
6521  PRINT PAGE
6530  COM Ar(26,101),Az(26,101),Ncode(26,101),Elem(900,11),Material(26,101)
6540  COM Nmel(100),Nemax(100),Nemin(100),Npmin(100),Npmax(100)
6550  COM Numtc,Elemax,Yield(10)
6560  COM Imin(100),Imax(100),Jmin(26),Jmax(26),Maxi,Maxj,Nmtl,Nbc
6570  COM Numnp,Numel,Nummat
6580  COM R(1000),Code(1000),Xr(1000),Z(1000),Xz(1000)
6590  COM Kp,Rp(250),Zp(250)
6600  COM Nele(26)
6610  COM Ilow,Ihigh,Jlow,Jhigh
6620  COM Npp,Nseg
6630  COM File$[15],Line$[81],Que$,P$
6640 Scale:!  ****** AUTOMATIC SCALING ******
6650  Ihflg=1
6660  Ilflg=1
6670  FOR K=Ilow TO Ihigh
6680  IF Ncode(K,Jhigh)=0 THEN GOTO 6710
6690  Zmax=MAX(Zmax,Az(K,Jhigh))
6700  Ihflg=0
6710  IF Ncode(K,Jlow)=0 THEN GOTO 6740
6720  Zmin=MIN(Zmin,Az(K,Jlow))
6730  Ilflg=0
6740  NEXT K
6750  IF Ihflg=1 THEN GOTO 6930
6760  IF Ilflg=1 THEN GOTO 6950
6770  Ihflg=1
6780  Ilflg=1
6790  FOR K=Jlow TO Jhigh
6800  IF Ncode(Ihigh,K)=0 THEN GOTO 6830
6810  Rmax=MAX(Rmax,Ar(Ihigh,K))
6820  Ihflg=0
6830  IF Ncode(Ilow,K)=0 THEN GOTO 6860
6840  Rmin=MIN(Rmin,Ar(Ilow,K))
6850  Ilflg=0
6860  NEXT K
6870  IF Ihflg=1 THEN GOTO 6970
6880  IF Ilflg=1 THEN GOTO 6990
6890  CALL Full(Zmax,A,B,D)
6900  IF Zmax<=Rmax/.9 THEN CALL Rscale(Rmax,A,B,D)
6910  SCALE A,B,0,D
6920  GOTO 7010
6930  Jhigh=Jhigh-1
6940  GOTO 6650
6950  Jlow=Jlow+1
6960  GOTO 6650
6970  Ihigh=Ihigh-1
6980  GOTO 6770
6990  Ilow=Ilow+1
7000  GOTO 6770
7010  ! ****** PLOT IN THE J-DIRECTION ******
7020  Movflg=1
7030  Jhm1=Jhigh
7040  INTEGER Ki,Kj
```

```
7050  FOR Ki=Ilow TO Ihigh
7060  FOR Kj=Jlow TO Jhm1
7070  Ndcode=Ncode(Ki,Kj)
7080  IF (Ndcode=3) OR (Ndcode=2) THEN GOTO 7110
7090  Movflg=1
7100  GOTO 7230
7110  IF Movflg=1 THEN GOTO 7160
7120  Xzj=Az(Ki,Kj+1)
7130  Yri=Ar(Ki,Kj+1)
7140  DRAW Xzj,Yri
7150  GOTO 7230
7160  Movflg=0
7170  Xzj=Az(Ki,Kj)
7180  Yri=Ar(Ki,Kj)
7190  MOVE Xzj,Yri
7200  Xzj=Az(Ki,Kj+1)
7210  Yri=Ar(Ki,Kj+1)
7220  DRAW Xzj,Yri
7230  NEXT Kj
7240  Movflg=1
7250  NEXT Ki
7260  ! ****** PLOT IN THE I-DIRECTION ******
7270  FOR Kj=Jlow TO Jhigh
7280  Ihm1=Ihigh
7290  FOR Ki=Ilow TO Ihm1
7300  Ndcode=Ncode(Ki,Kj)
7310  IF (Ndcode=3) OR (Ndcode=1) THEN GOTO 7340
7320  Movflg=1
7330  GOTO 7460
7340  IF Movflg=1 THEN GOTO 7390
7350  Xzj=Az(Ki+1,Kj)
7360  Yri=Ar(Ki+1,Kj)
7370  DRAW Xzj,Yri
7380  GOTO 7460
7390  Movflg=0
7400  Xzj=Az(Ki,Kj)
7410  Yri=Ar(Ki,Kj)
7420  MOVE Xzj,Yri
7430  Xzj=Az(Ki+1,Kj)
7440  Yri=Ar(Ki+1,Kj)
7450  DRAW Xzj,Yri
7460  NEXT Ki
7470  Movflg=1
7480  NEXT Kj
7490  IF Ans$="Y" THEN OUTPUT 7,5;"SP0"
7500  IF Ans$="Y" THEN PLOTTER 7,5 IS OFF
7510  IF R$="N" THEN GOTO 7580
7520  PLOTTER Grid(*) IS OFF
7530  INPUT "Enter name of grid file: G      ",Gfile$
7540  Gfile$=Gfile$&":C12"
7550  CREATE Gfile$,300
7560  ASSIN #2 TO Gfile$
7570  PRINT #2;Grid(*)
7580              DIM Element(100,3)
7590              INTEGER Materiol
7600              DIM Materiol(Maxi,Maxj)
7610              PRINT PAGE
7620              PRINT "When you are finished viewing the grid press  C "
7630              PRINT "                                              O "
7640              PRINT "                                              N "
7650              PRINT "                                              T "
7660              PAUSE
7670              GCLEAR
7680 Pre:         PRINT PAGE
7690              IF Que$="Y" THEN PRINTER IS 0
7700              FOR Mat=1 TO Nmtl
```

```
7710              READ #1;Line$
7720              Hue=VAL(Line$[1,5])
7730              I1=VAL(Line$[6,10])
7740              Ih=VAL(Line$[11,15])
7750              J1=VAL(Line$[16,20])
7760              Jh=VAL(Line$[21,25])
7770              IF Que$="Y" THEN PRINT USING Image3;Hue,I1,Ih,J1,Jh
7780 Image3:      IMAGE 5(5D)
7790              FOR J=J1 TO Jh-1
7800              FOR I=I1 TO Ih-1
7810              IF R$="S" THEN GOTO 7830
7820              IF P$="A" THEN GOTO 7830
7830              Material(I,J)=Hue
7840              Materiol(I,J)=Hue
7850              PLOTTER IS Element(*)
7860              MOVE Az(I,J),Ar(I,J)
7870              PEN 1
7880              DRAW Az(I+1,J),Ar(I+1,J)
7890              DRAW Az(I+1,J+1),Ar(I+1,J+1)
7900              DRAW Az(I,J+1),Ar(I,J+1)
7910              DRAW Az(I,J),Ar(I,J)
7920              PLOTTER Element(*) IS OFF
7930              Huey=Hue
7940              IF Hue>6 THEN Huey=Hue-4.5
7950              AREA COLOR Huey/6,1,1
7960              MAT PLOT Element,FILL
8030              NEXT I
8040              NEXT J
8050              NEXT Mat
8060              PRINTER IS 16
8070              INPUT "Do you want to create a Supergrid File?(Y/N)",R$
8080              IF R$="Y" THEN GOTO Supergrid
8090              GOTO 8560
8100 Supergrid:!
8110 !
8120 ! ********* READ MATERIAL INFORMATION CARDS **********
8130 !
8140          READ #1;Line$
8150          FOR I=1 TO Nummat
8160          READ #1;Line$
8170          Numid=VAL(Line$[1,5])
8180          READ #1;Line$
8190          READ #1;Line$
8200          Yield(Numid)=VAL(Line$[31,40])
8210          NEXT I
8220          FOR J=1 TO Maxj
8230          FOR I=1 TO Maxi
8240          IF Material(I,J)=0 THEN GOTO 8460
8250 !
8260 !            **** Exchange mat'l # for Yield Strength ****
8270 !
8280              FOR L=1 TO 10
8290              IF Material(I,J)=L THEN Material(I,J)=Yield(L)
8300              NEXT L
8310 !
8320 !         **** Create Element Information Array ****
8330 !
8340          Elnum=Elnum+1
8350          Elem(Elnum,1)=Elnum!.................Element #
8360          Elem(Elnum,2)=Material(I,J)!..........Material Yield Strength
8370          Elem(Elnum,3)=Az(I,J)!................LL-z
8380          Elem(Elnum,4)=Ar(I,J)!................LL-r
8390          Elem(Elnum,5)=Az(I+1,J)!..............UL-z
8400          Elem(Elnum,6)=Ar(I+1,J)!..............UL-r     UL__UR
8410          Elem(Elnum,7)=Az(I+1,J+1)!............UR-z       |__|<---Element
8420          Elem(Elnum,8)=Ar(I+1,J+1)!............UR-r     LL   LR
```

```
8430            Elem(Elnum,9)=Az(I,J+1)!..............LR-z
8440            Elem(Elnum,10)=Ar(I,J+1)!.............LR-r
8450            Elem(Elnum,11)=Materiol(I,J)!.........Material I.D.#
8460            NEXT I
8470            NEXT J
8480            Elemax=Elnum
8490            IF R$="N" THEN GOTO 8580
8500            INPUT "Enter name of Supergrid file: S_____",Sfile$
8510            Sfile$=Sfile$&":C12"
8520            CREATE Sfile$,Elemax*11/4+1
8530            ASSIGN #2 TO Sfile$
8540            PRINT #2;Elemax
8550            MAT PRINT #2;Elem
8560                  BEEP
8570                  PRINTER IS 16
8580                  PRINT "Would you like a nodal point listing?(Y/N)"
8590                  INPUT A$
8600                  IF A$="N" THEN GOTO 8780
8610                  INPUT "Does the grid contain the I,J origin?(Y/N)",D$
8620                  PRINTER IS 0
8630                  PRINT USING Image1
8640 Image1:          IMAGE 3X"I",3X"J",7X"R",10X"Z",10X"NP"
8650                  FOR K=1 TO Maxj
8660                  FOR L=1 TO Maxi
8670                  IF (D$="N") AND (K=1) AND (L=1) THEN GOTO 8710
8680                  IF ((K<>1) OR (L<>1)) AND (Ar(L,K)=0) AND (Az(L,K)=0) THEN G
OTO 8710
8690                  Jim=Jim+1
8700                  PRINT USING Image2;L,K,Ar(L,K),Az(L,K),Jim
8710 Image2:          IMAGE 1X,DDD,1X,DDD,5X,DD.DDD,5X,DD.DDD,5X,DDDD
8720                  NEXT L
8730                  NEXT K
8740                  BEEP
8750                  WAIT 200
8760                  BEEP
8770                  WAIT 200
8780                  BEEP
8790                  SUBEXIT
8800                  SUBEND
8810        SUB Circles(Ang1,Delphi,Rstrt,Zstrt,Rc,Zc,I,J)
8820        OPTION BASE 1
8830        INTEGER Ncode,Material,Imin,Imax,Jmin,Jmax,Code
8840        COM Ar(26,101),Az(26,101),Ncode(26,101),Elem(900,11),Material(26,101)
8850        COM Nmel(100),Nemax(100),Nemin(100),Npmin(100),Npmax(100)
8860        COM Numtc,Elemax,Yield(10)
8870        COM Imin(100),Imax(100),Jmin(26),Jmax(26),Maxi,Maxj,Nmtl,Nbc
8880        COM Numnp,Numel,Nummat
8890        COM R(1000),Code(1000),Xr(1000),Z(1000),Xz(1000)
8900        COM Kp,Rp(250),Zp(250)
8910        COM Nele(26)
8920        COM Ilow,Ihigh,Jlow,Jhigh
8930        COM Npp,Nseg
8940        COM File$[15],Line$[81],Que$,P$
8950        ! ********************************************** !
8960        !  FIND INTERSECTION OF LINE & CIRCLE = NEW R & Z
8970        ! ********************************************** !
8980        Ang1=Ang1+Delphi
8990        Rr=SQR((Rstrt-Rc)^2+(Zstrt-Zc)^2)
9000        Ar(I,J)=Rc+Rr*COS(Ang1)
9010        Az(I,J)=Zc+Rr*SIN(Ang1)
9020        SUBEXIT
9030        SUBEND
9040            SUB Full(Zmax,A,B,D)
9050            A=Zmin
9060            B=Zmax
9070            D=Zmax/1.23
```

```
9080            SUBEXIT
9090            SUBEND
9100          SUB Rscale(Rmax,A,B,D)
9110          A=Zmin
9120          B=Rmax*1.23*1.2
9130          D=Rmax*1.2
9140          SUBEXIT
9150          SUBEND
9160      SUB Grid1(#2,Gfile$,Grid)
9170      PLOTTER IS "GRAPHICS"
9180      GRAPHICS
9190      SUBEXIT
9200      SUBEND
9210      SUB Grid2(#2,Gfile$,Grid)
9220      INPUT "Enter name of grid file: G      ",Gfile$
9230      Gfile$=Gfile$&":C12"
9240      CREATE Gfile$,400
9250      ASSIGN #2 TO Gfile$
9260      DIM Grid(1500,3)
9270      PRINT #2;Grid(*)
9280      SUBEXIT
9290      SUBEND
```

```
1       ! ********* PROGRAM "PREP1" OF THE PREPPY PACKAGE **********
10      OPTION BASE 1
20      PRINTER IS 16
30      PLOTTER IS "GRAPHICS"
40      GRAPHICS
50      CALL Plots
100     END
110     SUB Plots
120     OPTION BASE 1
130     INPUT "Enter name of Supergrid file:",Sfile$
140     DISP "* Reading in Supergrid File: ";Sfile$
150     Sfile$=Sfile$&":C12"
160     ASSIGN #1 TO Sfile$
170     DIM Elem(850,11)
180     READ #1;Elemax
190     REDIM Elem(Elemax,11)
200     MAT READ #1;Elem
210     MAT SEARCH Elem(*,7),MAX;Zmax1
211     MAT SEARCH Elem(*,3),MIN;Zmin1
220     MAT SEARCH Elem(*,9),MAX;Zmax2
221     MAT SEARCH Elem(*,5),MIN;Zmin2
230     MAT SEARCH Elem(*,8),MAX;Rmax1
240     MAT SEARCH Elem(*,10),MAX;Rmax2
250     Zmax=MAX(Zmax1,Zmax2)
251     Zmin=MIN(Zmin1,Zmin2)
260     Rmax=MAX(Rmax1,Rmax2)
270             DIM Stress$(850)[80]
280             DIM Element(10,3)
290             SHORT Stress(850,7)
300             REDIM Stress(Elemax,7)
310             INPUT "Material plot or Stress plot?(M/S)",Choice$
320             IF Choice$="M" THEN GOTO Scalemenu
330             INPUT "Enter name of output file:",Out$
340             Out$=Out$&":C12"
350             ASSIGN #3 TO Out$
360             DISP "                          * Organizing Stress Array *"
370             READ #3;Stress$(1)
380             FOR I=1 TO Elemax
390             READ #3;Stress$(I)
400             Stress(I,1)=VAL(Stress$(I)[1,4])
410             Stress(I,2)=VAL(Stress$(I)[5,11])
420             Stress(I,3)=VAL(Stress$(I)[13,19])
430             Stress(I,4)=VAL(Stress$(I)[21,27])
440             Stress(I,5)=VAL(Stress$(I)[29,35])
450             Stress(I,6)=VAL(Stress$(I)[37,43])
460             Stress(I,7)=Elem(I,11)
470             NEXT I
480 Scalemenu:  !
490     PRINT "Scale Menu: FULL |           |"
500     PRINT "             1/2 |     |     |"
510     PRINT "             2/2 |     |     |"
520     PRINT "             1/4 |  |  |     |"
530     PRINT "             2/4 |  |  |     |"
540     PRINT "             3/4 |  |  |  |  |"
550     PRINT "             4/4 |  |  |  |  |"
560     INPUT Menu$
570     PRINT PAGE
580     GCLEAR
590     IF (Zmax>Rmax) AND (Menu$="FULL") THEN CALL Full(Zmax,A,B,D)
600     IF (Zmax>Rmax) AND (Menu$="1/2") THEN CALL One_half(Zmax,A,B,D)
610     IF (Zmax>Rmax) AND (Menu$="2/2") THEN CALL Two_half(Zmax,A,B,D)
620     IF (Zmax>Rmax) AND (Menu$="1/4") THEN CALL One_fourth(Zmax,A,B,D)
630     IF (Zmax>Rmax) AND (Menu$="2/4") THEN CALL Two_fourth(Zmax,A,B,D)
640     IF (Zmax>Rmax) AND (Menu$="3/4") THEN CALL Three_fourth(Zmax,A,B,D)
650     IF (Zmax>Rmax) AND (Menu$="4/4") THEN CALL Four_fourth(Zmax,A,B,D)
660     IF .99*Zmax<=Rmax THEN CALL Rscale(Rmax,A,B,D)
```

```
661             IF (Menu$="FULL") OR (Menu$="1/2") OR (Menu$="1/4") THEN A=Zmin
670             SCALE A,B,0,D
680             IF Choice$="M" THEN GOTO Effective
690             PRINT "Specify Stress Component: R,Z,T,RZ,EFF"
700             INPUT Dir$
701             IF (Menu$="FULL") OR (Menu$="1/2") OR (Menu$="1/4") THEN A=Zmin
710             IF Chartflag=1 THEN SCALE A,B,0,D
720             IF (Dir$<>"R") AND (Dir$<>"Z") AND (Dir$<>"T") AND (Dir$<>"RZ") AN
D (Dir$<>"EFF") THEN PRINT "INVALID ENTRY"
730             IF (Dir$<>"R") AND (Dir$<>"Z") AND (Dir$<>"T") AND (Dir$<>"RZ") AN
D (Dir$<>"EFF") THEN OUTPUT 7,6;"BP7,300,5;BP6,300,5"
740             IF (Dir$<>"R") AND (Dir$<>"Z") AND (Dir$<>"T") AND (Dir$<>"RZ") AN
D (Dir$<>"EFF") THEN GOTO 690
750             IF Dir$<>"EFF" THEN GOTO 790
760             PRINT PAGE
770             PRINT "Component = EFF"
780             GOTO Effective
790             PRINT "Specify Direction:(-/+)"
800             INPUT Ct$
810             PRINT PAGE
820             GCLEAR
830             PRINT "Component = ";Dir$,"Direction = ";Ct$
840             GOTO Other_direction
850 Effective:    IF Choice$="S" THEN CALL Chart
851               IF (Menu$="FULL") OR (Menu$="1/2") OR (Menu$="1/4") THEN A=Zmin
860               SCALE A,B,0,D
870               MAT SORT Stress(*,1)
880               FOR I=1 TO Elemax
900               Hue=Elem(I,11)
910               IF Hue=1 THEN 1240!***IF YOU WANT AIR PLOTTED COMMENT THIS LINE
915               IF Choice$="M" THEN GOTO 960
920               Intensity=Stress(I,6)/Elem(I,2)
930               Intense=Intensity
940               IF Intensity>1 THEN Intensity=1
950               IF Intensity<0 THEN Intensity=0
960               IF Hue>6 THEN Hue=Hue-4.5
970               Hue=Hue/6
980               PLOTTER IS Element(*)
990               MOVE Elem(I,3),Elem(I,4)
1000              PEN -1
1010              DRAW Elem(I,5),Elem(I,6)
1020              DRAW Elem(I,7),Elem(I,8)
1030              DRAW Elem(I,9),Elem(I,10)
1040              DRAW Elem(I,3),Elem(I,4)
1050              PLOTTER Element(*) IS OFF
1060              IF Choice$="M" THEN Intensity=1
1070              AREA COLOR Hue,Intensity,1
1080              MAT PLOT Element,FILL
1090              IF Choice$="S" THEN 1170
1100              MOVE Elem(I,3),Elem(I,4)
1110              PEN -1
1120              DRAW Elem(I,5),Elem(I,6)
1130              DRAW Elem(I,7),Elem(I,8)
1140              DRAW Elem(I,9),Elem(I,10)
1150              DRAW Elem(I,3),Elem(I,4)
1160 !
1170 !          ***** ELEMENTS ABOVE YIELD *****
1180 !
1190            IF (Intense<1.02) OR (Dir$<>"EFF") THEN GOTO 1240
1200            MOVE Elem(I,3),Elem(I,4)
1210            DRAW Elem(I,7),Elem(I,8)
1220            MOVE Elem(I,5),Elem(I,6)
1230            DRAW Elem(I,9),Elem(I,10)
1240            NEXT I
1250 !
1260 !          *********************************
```

```
1270 !
1280          PRINTER IS 16
1290          IF Choice$="M" THEN INPUT "Look at another section?(Y/N)",Lo
ok$
1300          IF Look$="N" THEN 3270
1310          IF Choice$="M" THEN GOTO Scalemenu
1320          INPUT "More data (Y/N)",Huh$
1330          IF Huh$="N" THEN GOTO 3270
1340          PRINT PAGE
1350          PRINT "Stress Component or Output File: SC/OP"
1360          INPUT An$
1370          IF (An$<>"SC") AND (An$<>"OP") THEN PRINT "INVALID ENTRY"
1380          IF (An$<>"SC") AND (An$<>"OP") THEN GOTO 1350
1390          PRINT PAGE
1400          IF An$="SC" THEN GOTO Scalemenu
1410          IF An$="OP" THEN GOTO 330
1420 Other_direction:!
1440     DIM One(350,5),Two(850,5),Three(300,5),Four(850,5),Five(300,5),Six(850,
5),Seven(850,5),Eight(300,5),Nine(300,5),Ten(850,5),Maxvals(10,8)
1450     K1=K2=K3=K4=K5=K6=K7=K8=K9=K10=0
1460     MAT SORT Stress(*,7) DES
1470     Loc=Elemax+1
1480     FOR Mat1=1 TO 10
1490     MAT SEARCH Stress(*,7),LOC(=Mat1);Oldloc
1500     IF Oldloc=Elemax+1 THEN 1520
1510     ON Stress(Loc-1,7) GOTO One,Two,Three,Four,Five,Six,Seven,Eight,Nine,Te
n
1520 !   CONTINUE
1530     NEXT Mat1
1540     GOTO 3140
1550 One:! ************************************
1560       REDIM One(Loc-Oldloc,5)
1570       FOR I=Oldloc TO Loc-1
1580       K1=K1+1
1590       FOR J=1 TO 5
1600       One(K1,J)=Stress(I,J)
1610       NEXT J
1620       NEXT I
1630       Loc=Oldloc
1640       FOR K=1 TO 4
1650       MAT SEARCH One(*,K+1),MAX;Maxvals(1,K)
1660       MAT SEARCH One(*,K+1),MIN;Maxvals(1,K+4)
1670       NEXT K
1680       Maxvalmark=1
1690       CALL Directionplot(Elem(*),One(*),K1,Dir$,Ct$,Maxvalmark,Maxvals(*))
1700       GOTO 1520
1710 Two: ! ************************************
1720       REDIM Two(Loc-Oldloc,5)
1730       FOR I=Oldloc TO Loc-1
1740       K2=K2+1
1750       FOR J=1 TO 5
1760       Two(K2,J)=Stress(I,J)
1770       NEXT J
1780       NEXT I
1790       Loc=Oldloc
1800       FOR K=1 TO 4
1810       MAT SEARCH Two(*,K+1),MAX;Maxvals(2,K)
1820       MAT SEARCH Two(*,K+1),MIN;Maxvals(2,K+4)
1830       NEXT K
1840       Maxvalmark=2
1850       CALL Directionplot(Elem(*),Two(*),K2,Dir$,Ct$,Maxvalmark,Maxvals(*))
1860       GOTO 1520
1870 Three:! ************************************
1880       REDIM Three(Loc-Oldloc,5)
1890       FOR I=Oldloc TO Loc-1
1900       K3=K3+1
```

```
1910        FOR J=1 TO 5
1920        Three(K3,J)=Stress(I,J)
1930        NEXT J
1940        NEXT I
1950        Loc=Oldloc
1960        FOR K=1 TO 4
1970        MAT SEARCH Three(*,K+1),MAX;Maxvals(3,K)
1980        MAT SEARCH Three(*,K+1),MIN;Maxvals(3,K+4)
1990        NEXT K
2000        Maxvalmark=3
2010        CALL Directionplot(Elem(*),Three(*),K3,Dir$,Ct$,Maxvalmark,Maxvals(*)
)
2020        GOTO 1520
2030 Four:! ************************************
2040        REDIM Four(Loc-Oldloc,5)
2050        FOR I=Oldloc TO Loc-1
2060        K4=K4+1
2070        FOR J=1 TO 5
2080        Four(K4,J)=Stress(I,J)
2090        NEXT J
2100        NEXT I
2110        Loc=Oldloc
2120        FOR K=1 TO 4
2130        MAT SEARCH Four(*,K+1),MAX;Maxvals(4,K)
2140        MAT SEARCH Four(*,K+1),MIN;Maxvals(4,K+4)
2150        NEXT K
2160        Maxvalmark=4
2170        CALL Directionplot(Elem(*),Four(*),K4,Dir$,Ct$,Maxvalmark,Maxvals(*))
2180        GOTO 1520
2190 Five:! ************************************
2200        REDIM Five(Loc-Oldloc,5)
2210        FOR I=Oldloc TO Loc-1
2220        K5=K5+1
2230        FOR J=1 TO 5
2240        Five(K5,J)=Stress(I,J)
2250        NEXT J
2260        NEXT I
2270        Loc=Oldloc
2280        FOR K=1 TO 4
2290        MAT SEARCH Five(*,K+1),MAX;Maxvals(5,K)
2300        MAT SEARCH Five(*,K+1),MIN;Maxvals(5,K+4)
2310        NEXT K
2320        Maxvalmark=5
2330        CALL Directionplot(Elem(*),Five(*),K5,Dir$,Ct$,Maxvalmark,Maxvals(*))
2340        GOTO 1520
2350 Six:! *************************************
2360        REDIM Six(Loc-Oldloc,5)
2370        FOR I=Oldloc TO Loc-1
2380        K6=K6+1
2390        FOR J=1 TO 5
2400        Six(K6,J)=Stress(I,J)
2410        NEXT J
2420        NEXT I
2430        Loc=Oldloc
2440        FOR K=1 TO 4
2450        MAT SEARCH Six(*,K+1),MAX;Maxvals(6,K)
2460        MAT SEARCH Six(*,K+1),MIN;Maxvals(6,K+4)
2470        NEXT K
2480        Maxvalmark=6
2490        CALL Directionplot(Elem(*),Six(*),K6,Dir$,Ct$,Maxvalmark,Maxvals(*))
2500        GOTO 1520
2510 Seven:! ***********************************
2520        REDIM Seven(Loc-Oldloc,5)
2530        FOR I=Oldloc TO Loc-1
2540        K7=K7+1
2550        FOR J=1 TO 5
```

```
2560        Seven(K7,J)=Stress(I,J)
2570        NEXT J
2580        NEXT I
2590        Loc=Oldloc
2600        FOR K=1 TO 4
2610        MAT SEARCH Seven(*,K+1),MAX;Maxvals(7,K)
2620        MAT SEARCH Seven(*,K+1),MIN;Maxvals(7,K+4)
2630        NEXT K
2640        Maxvalmark=7
2650        CALL Directionplot(Elem(*),Seven(*),K7,Dir$,Ct$,Maxvalmark,Maxvals(*)
)
2660        GOTO 1520
2670 Eight:! ************************************
2680        REDIM Eight(Loc-Oldloc,5)
2690        FOR I=Oldloc TO Loc-1
2700        K8=K8+1
2710        FOR J=1 TO 5
2720        Eight(K8,J)=Stress(I,J)
2730        NEXT J
2740        NEXT I
2750        Loc=Oldloc
2760        FOR K=1 TO 4
2770        MAT SEARCH Eight(*,K+1),MAX;Maxvals(8,K)
2780        MAT SEARCH Eight(*,K+1),MIN;Maxvals(8,K+4)
2790        NEXT K
2800        Maxvalmark=8
2810        CALL Directionplot(Elem(*),Eight(*),K8,Dir$,Ct$,Maxvalmark,Maxvals(*)
)
2820        GOTO 1520
2830 Nine:! *************************************
2840        REDIM Nine(Loc-Oldloc,5)
2850        FOR I=Oldloc TO Loc-1
2860        K9=K9+1
2870        FOR J=1 TO 5
2880        Nine(K9,J)=Stress(I,J)
2890        NEXT J
2900        NEXT I
2910        Loc=Oldloc
2920        FOR K=1 TO 4
2930        MAT SEARCH Nine(*,K+1),MAX;Maxvals(9,K)
2940        MAT SEARCH Nine(*,K+1),MIN;Maxvals(9,K+4)
2950        NEXT K
2960        Maxvalmark=9
2970        CALL Directionplot(Elem(*),Nine(*),K9,Dir$,Ct$,Maxvalmark,Maxvals(*))
2980        GOTO 1520
2990 Ten:! **************************************
3000        REDIM Ten(Loc-Oldloc,5)
3010        FOR I=Oldloc TO Loc-1
3020        K10=K10+1
3030        FOR J=1 TO 5
3040        Ten(K10,J)=Stress(I,J)
3050        NEXT J
3060        NEXT I
3070        FOR K=1 TO 4
3080        MAT SEARCH Ten(*,K+1),MAX;Maxvals(10,K)
3090        MAT SEARCH Ten(*,K+1),MIN;Maxvals(10,K+4)
3100        NEXT K
3110        Maxvalmark=10
3120        CALL Directionplot(Elem(*),Ten(*),K10,Dir$,Ct$,Maxvalmark,Maxvals(*))
3130        GOTO 1520
3140        CALL Chart2(Maxvals(*),Dir$,Ct$,Chartflag)
3150        MAT SORT Stress(*,1)
3160 ! ***************************************************************************
3170        INPUT "More Data?(Y/N)",Huh$
3180        IF Huh$="N" THEN GOTO 3270
3190        PRINT PAGE
```

```
3200        PRINT "Stress Component or Output File?(SC/OP)"
3210        INPUT An$
3220        IF (An$<>"SC") AND (An$<>"OP") THEN PRINT "INVALID ENTRY"
3230        IF (An$<>"SC") AND (An$<>"OP") THEN GOTO 3200
3240        PRINT PAGE
3250        IF An$="SC" THEN GOTO Scalemenu
3260        IF An$="OP" THEN GOTO 330
3270        SUBEXIT
3280        SUBEND
3290      SUB Chart
3300      PLOTTER IS "GRAPHICS"
3310      GRAPHICS
3320      FOR I=18 TO 94 STEP 4
3330      MOVE I,99
3340      AREA COLOR 6/6,(I-18)/76,1
3350      RECTANGLE 4,1,FILL
3360      NEXT I
3370      FOR I=18 TO 94 STEP 4
3380      MOVE I,98
3390      AREA COLOR 4/6,(I-18)/76,1
3400      RECTANGLE 4,1,FILL
3410      NEXT I
3420      FOR I=18 TO 94 STEP 4
3430      MOVE I,97
3440      AREA COLOR 2/6,(I-18)/76,1
3450      RECTANGLE 4,.8,FILL
3460      NEXT I
3470      MOVE 0,98
3480      PEN 1
3490      CSIZE 2.5,.6
3500      LABEL "ZERO STRESS"
3510      MOVE 100,98
3520      LABEL "YIELD STRESS"
3530      CSIZE 2.5,1
3540      LORG 5
3550      MOVE 61,94
3560      LABEL "EFFECTIVE STRESS"
3570      LORG 1
3580      PRINT PAGE
3590      SUBEXIT
3600      SUBEND
3610        SUB Full(Zmax,A,B,D)
3620        A=Zmin
3630        B=Zmax
3640        D=Zmax/1.23
3650        SUBEXIT
3660        SUBEND
3670          SUB One_half(Zmax,A,B,D)
3680          A=Zmin
3690          B=Zmax/2
3700          D=Zmax/1.23/2
3710          SUBEXIT
3720          SUBEND
3730        SUB Two_half(Zmax,A,B,D)
3740        A=Zmax/2
3750        B=Zmax
3760        D=Zmax/1.23/2
3770        SUBEXIT
3780        SUBEND
3790          SUB One_fourth(Zmax,A,B,D)
3800          A=Zmin
3810          B=Zmax/4
3820          D=Zmax/1.23/4
3830          SUBEXIT
3840          SUBEND
3850        SUB Two_fourth(Zmax,A,B,D)
```

```
3860        A=Zmax/4
3870        B=Zmax/2
3880        D=Zmax/1.23/4
3890        SUBEXIT
3900        SUBEND
3910          SUB Three_fourth(Zmax,A,B,D)
3920          A=Zmax/2
3930          B=3*Zmax/4
3940          D=Zmax/1.23/4
3950          SUBEXIT
3960          SUBEND
3970        SUB Four_fourth(Zmax,A,B,D)
3980        A=3*Zmax/4
3990        B=Zmax
4000        D=Zmax/1.23/4
4010        SUBEXIT
4020        SUBEND
4030          SUB Rscale(Rmax,A,B,D)
4040          A=0
4050          B=Rmax*1.23*1.2
4060          D=Rmax*1.2
4070          SUBEXIT
4080          SUBEND
4090    SUB Directionplot(Elem(*),Array(*),Kount,Dir$,Ct$,Maxvalmark,Maxvals(*))
4100    OPTION BASE 1
4110    DIM Element(50,3)
4120    IF Dir$="R" THEN Dir=1
4130    IF Dir$="Z" THEN Dir=2
4140    IF Dir$="T" THEN Dir=3
4150    IF Dir$="RZ" THEN Dir=4
4160    Olddir=Dir
4170    IF Ct$="-" THEN Dir=Dir+4
4180    FOR I=1 TO Kount
4190    Elnum=Array(I,1)
4200    Hue=Elem(Elnum,11)
4210    IF Hue=1 THEN 4430
4220    IF Hue>6 THEN Hue=Hue-4.5
4230    Hue=Hue/6
4240    IF Maxvals(Maxvalmark,Dir)=0 THEN GOTO 4290
4250    Intensity=Array(I,Olddir+1)/Maxvals(Maxvalmark,Dir)
4260    IF (Array(I,Olddir+1)<0) AND (Ct$="+") THEN Intensity=0
4270    IF (Array(I,Olddir+1)>0) AND (Ct$="-") THEN Intensity=0
4280    GOTO 4300
4290    Intensity=0
4300    PLOTTER IS Element(*)
4310    Luminosity=1
4320    IF Intensity=0 THEN Luminosity=.15
4330    IF Intensity=0 THEN Intensity=.90
4340    MOVE Elem(Elnum,3),Elem(Elnum,4)
4350    PEN 1
4360    DRAW Elem(Elnum,5),Elem(Elnum,6)
4370    DRAW Elem(Elnum,7),Elem(Elnum,8)
4380    DRAW Elem(Elnum,9),Elem(Elnum,10)
4390    DRAW Elem(Elnum,3),Elem(Elnum,4)
4400    PLOTTER Element(*) IS OFF
4410    AREA COLOR Hue,Intensity,Luminosity
4420    MAT PLOT Element,FILL
4430    NEXT I
4440    SUBEXIT
4450    SUBEND
4460    SUB Chart2(Maxvals(*),Dir$,Ct$,Chartflag)
4470    OPTION BASE 1
4480    PRINT PAGE
4490    SCALE 0,123,0,100
4500    IF Dir$="R" THEN Index=1
4510    IF Dir$="Z" THEN Index=2
```

```
4520    IF Dir$="T" THEN Index=3
4530    IF Dir$="RZ" THEN Index=4
4540    IF Ct$="-" THEN Index=Index+4
4550    MOVE 62,94
4560    CSIZE 2.5,1
4570    LORG 5
4580    IF (Ct$="-") AND (Dir$="R") THEN LABEL "COMPRESSIVE RADIAL STRESS"
4590    IF (Ct$="-") AND (Dir$="Z") THEN LABEL "COMPRESSIVE AXIAL STRESS"
4600    IF (Ct$="-") AND (Dir$="T") THEN LABEL "COMPRESSIVE HOOP STRESS"
4610    IF (Ct$="-") AND (Dir$="RZ") THEN LABEL "NEGATIVE SHEAR STRESS"
4620    IF (Ct$="+") AND (Dir$="R") THEN LABEL "TENSILE RADIAL STRESS"
4630    IF (Ct$="+") AND (Dir$="Z") THEN LABEL "TENSILE AXIAL STRESS"
4640    IF (Ct$="+") AND (Dir$="T") THEN LABEL "TENSILE HOOP STRESS"
4650    IF (Ct$="+") AND (Dir$="RZ") THEN LABEL "POSITIVE SHEAR STRESS"
4660    LORG 1
4670    FOR I=1 TO 10
4680    Maxval=Maxvals(I,Index)
4690    IF (Ct$="-") AND (Maxval>0) THEN Maxval=0
4700    IF (Ct$="+") AND (Maxval<0) THEN Maxval=0
4710    Hue=I
4720    IF Hue>6 THEN Hue=Hue-4.5
4730    Hue=Hue/6
4740    MOVE (I-1)*12.4,98
4750    AREA COLOR Hue,1,1
4760    RECTANGLE 2,2,FILL
4770    MOVE (I-1)*12.4+2.2,98
4780    PEN 1
4790    CSIZE 2.8,1/2
4800    LABEL Maxval
4810    NEXT I
4820    Chartflag=1
4830    SUBEXIT
4840    SUBEND
```

```
1     ! ******** PROGRAM "PREP2" OF THE PREPPY PACKAGE ********
2     ! THIS IS A BLACK & WHITE VERSION OF PREP1
10    OPTION BASE 1
20    PRINTER IS 16
30    PLOTTER IS "GRAPHICS"
40    GRAPHICS
50    CALL Plots
60    END
70    SUB Plots
80    OPTION BASE 1
90    INPUT "Enter name of Supergrid file:",Sfile$
100   DISP "* Reading in Supergrid File: ";Sfile$
110   Sfile$=Sfile$&":C12"
120   ASSIGN #1 TO Sfile$
130   DIM Elem(800,11)
140   READ #1;Elemax
150   REDIM Elem(Elemax,11)
160   MAT READ #1;Elem
170   MAT SEARCH Elem(*,7),MAX;Zmax1
180   MAT SEARCH Elem(*,9),MAX;Zmax2
190   MAT SEARCH Elem(*,8),MAX;Rmax1
200   MAT SEARCH Elem(*,10),MAX;Rmax2
210   Zmax=MAX(Zmax1,Zmax2)
220   Rmax=MAX(Rmax1,Rmax2)
230           DIM Stress$(800)[80]
240           DIM Element(10,3),Stress(800,7)
250           REDIM Stress(Elemax,7)
260           INPUT "Material plot or Stress plot?(M/S)",Choice$
270           IF Choice$="M" THEN GOTO Scalemenu
280           INPUT "Enter name of output file:",Out$
290           Out$=Out$&":C12"
300           ASSIGN #3 TO Out$
310           DISP "                          * Organizing Stress Array *"
320           READ #3;Stress$(1)
330           FOR I=1 TO Elemax
340           READ #3;Stress$(I)
350           Stress(I,1)=VAL(Stress$(I)[1,4])
360           Stress(I,2)=VAL(Stress$(I)[5,11])
370           Stress(I,3)=VAL(Stress$(I)[13,19])
380           Stress(I,4)=VAL(Stress$(I)[21,27])
390           Stress(I,5)=VAL(Stress$(I)[29,35])
400           Stress(I,6)=VAL(Stress$(I)[37,43])
410           Stress(I,7)=Elem(I,11)
420           NEXT I
430 Scalemenu:  !
440   PRINT "Scale Menu: FULL  |           |"
450   PRINT "             1/2  |     |     |"
460   PRINT "             2/2  |  |  |  |  |"
470   PRINT "             1/4  |  |  |  |  |"
480   PRINT "             2/4  |  |  |  |  |"
490   PRINT "             3/4  |  |  |  |  |"
500   PRINT "             4/4  |  |  |  |  |"
510   INPUT Menu$
520   PRINT PAGE
530   GCLEAR
540   IF (Zmax>Rmax) AND (Menu$="FULL") THEN CALL Full(Zma  ,B,D)
550   IF (Zmax>Rmax) AND (Menu$="1/2") THEN CALL One_half(  ax,A,B,D)
560   IF (Zmax>Rmax) AND (Menu$="2/2") THEN CALL Two_half  max,A,B,D)
570   IF (Zmax>Rmax) AND (Menu$="1/4") THEN CALL One_fou  h(Zmax,A,B,D)
580   IF (Zmax>Rmax) AND (Menu$="2/4") THEN CALL Two_fo  th(Zmax,A,B,D)
590   IF (Zmax>Rmax) AND (Menu$="3/4") THEN CALL Three  ourth(Zmax,A,B,D)
600   IF (Zmax>Rmax) AND (Menu$="4/4") THEN CALL Four_fourth(Zmax,A,B,D)
610   IF .99*Zmax<=Rmax THEN CALL Rscale(Rmax,A,B,D)
620           SCALE A,B,0,D
630           IF Choice$="M" THEN GOTO Effective
640           PRINT "Specify Stress Component: R,Z,T,RZ,EFF"
```

```
650             INPUT Dir$
660             IF Chartflag=1 THEN SCALE A,B,0,D
670             IF (Dir$<>"R") AND (Dir$<>"Z") AND (Dir$<>"T") AND (Dir$<>"RZ") AN
D (Dir$<>"EFF") THEN PRINT "INVALID ENTRY"
680             IF (Dir$<>"R") AND (Dir$<>"Z") AND (Dir$<>"T") AND (Dir$<>"RZ") AN
D (Dir$<>"EFF") THEN OUTPUT 7,6;"BP7,300,5;BP6,300,5"
690             IF (Dir$<>"R") AND (Dir$<>"Z") AND (Dir$<>"T") AND (Dir$<>"RZ") AN
D (Dir$<>"EFF") THEN GOTO 640
700             IF Dir$<>"EFF" THEN GOTO 740
710             PRINT PAGE
720             PRINT "Component = EFF"
730             GOTO Effective
740             PRINT "Specify Direction:(-/+)"
750             INPUT Ct$
760             PRINT PAGE
770             GCLEAR
780             PRINT "Component = ";Dir$,"Direction = ";Ct$
790             GOTO Other_direction
800 Effective:    IF Choice$="S" THEN CALL Chart
810               SCALE A,B,0,D
820               MAT SORT Stress(*,1)
830               FOR I=1 TO Elemax
840               Hue=Elem(I,11)
850               IF Hue=1 THEN 1190!***IF YOU WANT AIR PLOTTED COMMENT THIS LINE
860               IF Choice$="M" THEN GOTO 910
870               Intensity=Stress(I,6)/Elem(I,2)
880               Intense=Intensity
890               IF Intensity>1 THEN Intensity=1
900               IF Intensity<0 THEN Intensity=0
910               IF Hue>6 THEN Hue=Hue-4.5
920               Hue=Hue/6
930               PLOTTER IS Element(*)
940               MOVE Elem(I,3),Elem(I,4)
950               PEN -1
960               DRAW Elem(I,5),Elem(I,6)
970               DRAW Elem(I,7),Elem(I,8)
980               DRAW Elem(I,9),Elem(I,10)
990               DRAW Elem(I,3),Elem(I,4)
1000              PLOTTER Element(*) IS OFF
1010              IF Choice$="M" THEN Intensity=Hue
1020              AREA COLOR 1,1,Intensity
1030              MAT PLOT Element,FILL
1040              IF Choice$="S" THEN 1120
1050              MOVE Elem(I,3),Elem(I,4)
1060              PEN -1
1070              DRAW Elem(I,5),Elem(I,6)
1080              DRAW Elem(I,7),Elem(I,8)
1090              DRAW Elem(I,9),Elem(I,10)
1100              DRAW Elem(I,3),Elem(I,4)
1110 !
1120 !          ***** ELEMENTS ABOVE YIELD *****
1130 !
1140            IF (Intense<1.02) OR (Dir$<>"EFF") THEN GOTO 1190
1150            MOVE Elem(I,3),Elem(I,4)
1160            DRAW Elem(I,7),Elem(I,8)
1170            MOVE Elem(I,5),Elem(I,6)
1180            DRAW Elem(I,9),Elem(I,10)
1190            NEXT I
1200 !
1210 !          ********************************
1220 !
1230            PRINTER IS 0
1231            INPUT "DUMP GRAPHICS? (Y/N)",Dg$
1232            IF Dg$="Y" THEN DUMP GRAPHICS
1233            IF Dg$="Y" THEN PRINT
1234            PRINTER IS 16
```

```
1240          IF Choice$="M" THEN INPUT "Look at another section?(Y/N)",Lo
ok$
1250          IF Look$="N" THEN 3210
1260          IF Choice$="M" THEN GOTO Scalemenu
1270          INPUT "More data (Y/N)",Huh$
1280          IF Huh$="N" THEN GOTO 3210
1290          PRINT PAGE
1300          PRINT "Stress Component or Output File: SC/OP"
1310          INPUT An$
1320          IF (An$<>"SC") AND (An$<>"OP") THEN PRINT "INVALID ENTRY"
1330          IF (An$<>"SC") AND (An$<>"OP") THEN GOTO 1300
1340          PRINT PAGE
1350          IF An$="SC" THEN GOTO Scalemenu
1360          IF An$="OP" THEN GOTO 280
1370 Other_direction:!
1380     DIM One(350,5),Two(850,5),Three(300,5),Four(850,5),Five(300,5),Six(850,
5),Seven(850,5),Eight(300,5),Nine(300,5),Ten(850,5),Maxvals(10,8)
1390     K1=K2=K3=K4=K5=K6=K7=K8=K9=K10=0
1400     MAT SORT Stress(*,7) DES
1410     Loc=Elemax+1
1420     FOR Mat1=1 TO 10
1430     MAT SEARCH Stress(*,7),LOC(=Mat1);Oldloc
1440     IF Oldloc=Elemax+1 THEN 1460
1450     ON Stress(Loc-1,7) GOTO One,Two,Three,Four,Five,Six,Seven,Eight,Nine,Te
n
1460 !   CONTINUE
1470     NEXT Mat1
1480     GOTO 3080
1490 One:! ************************************
1500       REDIM One(Loc-Oldloc,5)
1510       FOR I=Oldloc TO Loc-1
1520       K1=K1+1
1530       FOR J=1 TO 5
1540       One(K1,J)=Stress(I,J)
1550       NEXT J
1560       NEXT I
1570       Loc=Oldloc
1580       FOR K=1 TO 4
1590       MAT SEARCH One(*,K+1),MAX;Maxvals(1,K)
1600       MAT SEARCH One(*,K+1),MIN;Maxvals(1,K+4)
1610       NEXT K
1620       Maxvalmark=1
1630       CALL Directionplot(Elem(*),One(*),K1,Dir$,Ct$,Maxvalmark,Maxvals(*))
1640       GOTO 1460
1650 Two: ! ************************************
1660       REDIM Two(Loc-Oldloc,5)
1670       FOR I=Oldloc TO Loc-1
1680       K2=K2+1
1690       FOR J=1 TO 5
1700       Two(K2,J)=Stress(I,J)
1710       NEXT J
1720       NEXT I
1730       Loc=Oldloc
1740       FOR K=1 TO 4
1750       MAT SEARCH Two(*,K+1),MAX;Maxvals(2,K)
1760       MAT SEARCH Two(*,K+1),MIN;Maxvals(2,K+4)
1770       NEXT K
1780       Maxvalmark=2
1790       CALL Directionplot(Elem(*),Two(*),K2,Dir$,Ct$,Maxvalmark,Maxvals(*))
1800       GOTO 1460
1810 Three:! ************************************
1820       REDIM Three(Loc-Oldloc,5)
1830       FOR I=Oldloc TO Loc-1
1840       K3=K3+1
1850       FOR J=1 TO 5
1860       Three(K3,J)=Stress(I,J)
```

```
1870        NEXT J
1880        NEXT I
1890        Loc=Oldloc
1900        FOR K=1 TO 4
1910        MAT SEARCH Three(*,K+1),MAX;Maxvals(3,K)
1920        MAT SEARCH Three(*,K+1),MIN;Maxvals(3,K+4)
1930        NEXT K
1940        Maxvalmark=3
1950        CALL Directionplot(Elem(*),Three(*),K3,Dir$,Ct$,Maxvalmark,Maxvals(*)
)
1960        GOTO 1460
1970 Four:! ************************************
1980        REDIM Four(Loc-Oldloc,5)
1990        FOR I=Oldloc TO Loc-1
2000        K4=K4+1
2010        FOR J=1 TO 5
2020        Four(K4,J)=Stress(I,J)
2030        NEXT J
2040        NEXT I
2050        Loc=Oldloc
2060        FOR K=1 TO 4
2070        MAT SEARCH Four(*,K+1),MAX;Maxvals(4,K)
2080        MAT SEARCH Four(*,K+1),MIN;Maxvals(4,K+4)
2090        NEXT K
2100        Maxvalmark=4
2110        CALL Directionplot(Elem(*),Four(*),K4,Dir$,Ct$,Maxvalmark,Maxvals(*))
2120        GOTO 1460
2130 Five:! ************************************
2140        REDIM Five(Loc-Oldloc,5)
2150        FOR I=Oldloc TO Loc-1
2160        K5=K5+1
2170        FOR J=1 TO 5
2180        Five(K5,J)=Stress(I,J)
2190        NEXT J
2200        NEXT I
2210        Loc=Oldloc
2220        FOR K=1 TO 4
2230        MAT SEARCH Five(*,K+1),MAX;Maxvals(5,K)
2240        MAT SEARCH Five(*,K+1),MIN;Maxvals(5,K+4)
2250        NEXT K
2260        Maxvalmark=5
2270        CALL Directionplot(Elem(*),Five(*),K5,Dir$,Ct$,Maxvalmark,Maxvals(*))
2280        GOTO 1460
2290 Six:! *************************************
2300        REDIM Six(Loc-Oldloc,5)
2310        FOR I=Oldloc TO Loc-1
2320        K6=K6+1
2330        FOR J=1 TO 5
2340        Six(K6,J)=Stress(I,J)
2350        NEXT J
2360        NEXT I
2370        Loc=Oldloc
2380        FOR K=1 TO 4
2390        MAT SEARCH Six(*,K+1),MAX;Maxvals(6,K)
2400        MAT SEARCH Six(*,K+1),MIN;Maxvals(6,K+4)
2410        NEXT K
2420        Maxvalmark=6
2430        CALL Directionplot(Elem(*),Six(*),K6,Dir$,Ct$,Maxvalmark,Maxvals(*))
2440        GOTO 1460
2450 Seven:! ***********************************
2460        REDIM Seven(Loc-Oldloc,5)
2470        FOR I=Oldloc TO Loc-1
2480        K7=K7+1
2490        FOR J=1 TO 5
2500        Seven(K7,J)=Stress(I,J)
2510        NEXT J
```

```
2520        NEXT I
2530        Loc=Oldloc
2540        FOR K=1 TO 4
2550        MAT SEARCH Seven(*,K+1),MAX;Maxvals(7,K)
2560        MAT SEARCH Seven(*,K+1),MIN;Maxvals(7,K+4)
2570        NEXT K
2580        Maxvalmark=7
2590        CALL Directionplot(Elem(*),Seven(*),K7,Dir$,Ct$,Maxvalmark,Maxvals(*)
)
2600        GOTO 1460
2610 Eight:! ************************************
2620        REDIM Eight(Loc-Oldloc,5)
2630        FOR I=Oldloc TO Loc-1
2640        K8=K8+1
2650        FOR J=1 TO 5
2660        Eight(K8,J)=Stress(I,J)
2670        NEXT J
2680        NEXT I
2690        Loc=Oldloc
2700        FOR K=1 TO 4
2710        MAT SEARCH Eight(*,K+1),MAX;Maxvals(8,K)
2720        MAT SEARCH Eight(*,K+1),MIN;Maxvals(8,K+4)
2730        NEXT K
2740        Maxvalmark=8
2750        CALL Directionplot(Elem(*),Eight(*),K8,Dir$,Ct$,Maxvalmark,Maxvals(*)
)
2760        GOTO 1460
2770 Nine:! *************************************
2780        REDIM Nine(Loc-Oldloc,5)
2790        FOR I=Oldloc TO Loc-1
2800        K9=K9+1
2810        FOR J=1 TO 5
2820        Nine(K9,J)=Stress(I,J)
2830        NEXT J
2840        NEXT I
2850        Loc=Oldloc
2860        FOR K=1 TO 4
2870        MAT SEARCH Nine(*,K+1),MAX;Maxvals(9,K)
2880        MAT SEARCH Nine(*,K+1),MIN;Maxvals(9,K+4)
2890        NEXT K
2900        Maxvalmark=9
2910        CALL Directionplot(Elem(*),Nine(*),K9,Dir$,Ct$,Maxvalmark,Maxvals(*))
2920        GOTO 1460
2930 Ten:! **************************************
2940        REDIM Ten(Loc-Oldloc,5)
2950        FOR I=Oldloc TO Loc-1
2960        K10=K10+1
2970        FOR J=1 TO 5
2980        Ten(K10,J)=Stress(I,J)
2990        NEXT J
3000        NEXT I
3010        FOR K=1 TO 4
3020        MAT SEARCH Ten(*,K+1),MAX;Maxvals(10,K)
3030        MAT SEARCH Ten(*,K+1),MIN;Maxvals(10,K+4)
3040        NEXT K
3050        Maxvalmark=10
3060        CALL Directionplot(Elem(*),Ten(*),K10,Dir$,Ct$,Maxvalmark,Maxvals(*))
3070        GOTO 1460
3080        CALL Chart2(Maxvals(*),Dir$,Ct$,Chartflag)
3090        MAT SORT Stress(*,1)
3100 ! ***************************************************************************
3110        INPUT "More Data?(Y/N)",Huh$
3120        IF Huh$="N" THEN GOTO 3210
3130        PRINT PAGE
3140        PRINT "Stress Component or Output File?(SC/OP)"
3150        INPUT An$
```

```
3160        IF (An$<>"SC") AND (An$<>"OP") THEN PRINT "INVALID ENTRY"
3170        IF (An$<>"SC") AND (An$<>"OP") THEN GOTO 3140
3180        PRINT PAGE
3190        IF An$="SC" THEN GOTO Scalemenu
3200        IF An$="OP" THEN GOTO 280
3210        SUBEXIT
3220        SUBEND
3230      SUB Chart
3240      PLOTTER IS "GRAPHICS"
3250      GRAPHICS
3260      FOR I=18 TO 94 STEP 4
3270      MOVE I,97
3280      AREA COLOR 1,1,(I-18)/76
3290      RECTANGLE 4,4,FILL
3300      NEXT I
3310      MOVE 0,98
3320      PEN 1
3330      CSIZE 2.5,.6
3340      LABEL "ZERO STRESS"
3350      MOVE 100,98
3360      LABEL "YIELD STRESS"
3370      CSIZE 2.5
3380      LORG 5
3390      MOVE 61,94
3400      LABEL "EFFECTIVE STRESS  X > 2% ABOVE YIELD"
3410      LORG 1
3420      PRINT PAGE
3430      SUBEXIT
3440      SUBEND
3450        SUB Full(Zmax,A,B,D)
3460        A=0
3470        B=Zmax
3480        D=Zmax/1.23
3490        SUBEXIT
3500        SUBEND
3510          SUB One_half(Zmax,A,B,D)
3520          A=0
3530          B=Zmax/2
3540          D=Zmax/1.23/2
3550          SUBEXIT
3560          SUBEND
3570        SUB Two_half(Zmax,A,B,D)
3580        A=Zmax/2
3590        B=Zmax
3600        D=Zmax/1.23/2
3610        SUBEXIT
3620        SUBEND
3630          SUB One_fourth(Zmax,A,B,D)
3640          A=0
3650          B=Zmax/4
3660          D=Zmax/1.23/4
3670          SUBEXIT
3680          SUBEND
3690        SUB Two_fourth(Zmax,A,B,D)
3700        A=Zmax/4
3710        B=Zmax/2
3720        D=Zmax/1.23/4
3730        SUBEXIT
3740        SUBEND
3750          SUB Three_fourth(Zmax,A,B,D)
3760          A=Zmax/2
3770          B=3*Zmax/4
3780          D=Zmax/1.23/4
3790          SUBEXIT
3800          SUBEND
3810        SUB Four_fourth(Zmax,A,B,D)
```

```
3820        A=3*Zmax/4
3830        B=Zmax
3840        D=Zmax/1.23/4
3850        SUBEXIT
3860        SUBEND
3870          SUB Rscale(Rmax,A,B,D)
3880          A=0
3890          B=Rmax*1.23*1.2
3900          D=Rmax*1.2
3910          SUBEXIT
3920          SUBEND
3930    SUB Directionplot(Elem(*),Array(*),Kount,Dir$,Ct$,Maxvalmark,Maxvals(*))
3940    OPTION BASE 1
3950    DIM Element(10,3)
3960    IF Dir$="R" THEN Dir=1
3970    IF Dir$="Z" THEN Dir=2
3980    IF Dir$="T" THEN Dir=3
3990    IF Dir$="RZ" THEN Dir=4
4000    Olddir=Dir
4010    IF Ct$="-" THEN Dir=Dir+4
4020    FOR I=1 TO Kount
4030    Elnum=Array(I,1)
4040    Hue=Elem(Elnum,11)
4050    IF Hue=1 THEN 4250
4060    IF Hue>6 THEN Hue=Hue-4.5
4070    Hue=Hue/6
4080    IF Maxvals(Maxvalmark,Dir)=0 THEN GOTO 4130
4090    Intensity=Array(I,Olddir+1)/Maxvals(Maxvalmark,Dir)
4100    IF (Array(I,Olddir+1)<0) AND (Ct$="+") THEN Intensity=0
4110    IF (Array(I,Olddir+1)>0) AND (Ct$="-") THEN Intensity=0
4120    GOTO 4140
4130    Intensity=0
4140    PLOTTER IS Element(*)
4150    MOVE Elem(Elnum,3),Elem(Elnum,4)
4160    PEN 1
4170    DRAW Elem(Elnum,5),Elem(Elnum,6)
4180    DRAW Elem(Elnum,7),Elem(Elnum,8)
4190    DRAW Elem(Elnum,9),Elem(Elnum,10)
4200    DRAW Elem(Elnum,3),Elem(Elnum,4)
4210    PLOTTER Element(*) IS OFF
4220    IF Intensity<.08 THEN Intensity=.08
4230    AREA COLOR 1,1,Intensity
4240    MAT PLOT Element,FILL
4250    NEXT I
4260    SUBEXIT
4270    SUBEND
4280    SUB Chart2(Maxvals(*),Dir$,Ct$,Chartflag)
4290    OPTION BASE 1
4300    PRINT PAGE
4310    SCALE 0,123,0,100
4320    IF Dir$="R" THEN Index=1
4330    IF Dir$="Z" THEN Index=2
4340    IF Dir$="T" THEN Index=3
4350    IF Dir$="RZ" THEN Index=4
4360    IF Ct$="-" THEN Index=Index+4
4370    MOVE 62,98.75
4380    CSIZE 2.5,1
4390    LORG 5
4400    IF (Ct$="-") AND (Dir$="R") THEN LABEL "COMPRESSIVE RADIAL STRESS"
4410    IF (Ct$="-") AND (Dir$="Z") THEN LABEL "COMPRESSIVE AXIAL STRESS"
4420    IF (Ct$="-") AND (Dir$="T") THEN LABEL "COMPRESSIVE HOOP STRESS"
4430    IF (Ct$="-") AND (Dir$="RZ") THEN LABEL "NEGATIVE SHEAR STRESS"
4440    IF (Ct$="+") AND (Dir$="R") THEN LABEL "TENSILE RADIAL STRESS"
4450    IF (Ct$="+") AND (Dir$="Z") THEN LABEL "TENSILE AXIAL STRESS"
4460    IF (Ct$="+") AND (Dir$="T") THEN LABEL "TENSILE HOOP STRESS"
4470    IF (Ct$="+") AND (Dir$="RZ") THEN LABEL "POSITIVE SHEAR STRESS"
```

```
4480    LORG 1
4490    MOVE 0,95
4500    CSIZE 2.525
4510    LABEL "MAT #1  MAT #2  MAT #3  MAT #4  MAT #5  MAT #6  MAT #7  MAT #8  MA
T #9  MAT #10"
4520    FOR I=0 TO 9
4530    Maxval=Maxvals(I+1,Index)
4540    IF (Ct$="-") AND (Maxval>0) THEN Maxval=0
4550    IF (Ct$="+") AND (Maxval<0) THEN Maxval=0
4560    MOVE I*12,92
4570    PEN 1
4580    CSIZE 2.8,1/2
4590    LABEL Maxval
4600    NEXT I
4610    INPUT "DUMP GRAPHICS? (Y/N)",Dg$
4611    PRINTER IS 0
4612    IF Dg$="Y" THEN PRINT
4620    IF Dg$="Y" THEN DUMP GRAPHICS
4621    PRINTER IS 16
4630    Chartflag=1
4640    SUBEXIT
4650    SUBEND
```

```
1       ! ******** PROGRAM "GRID" OF THE PREPPY PACKAGE ********
10      OPTION BASE 1
20      PRINTER IS 16
30      INPUT "Enter name of Grid file:",File$
40      Name$=File$
50      File$=File$&":C12"
60      DIM Plot(2000,3)
70      INPUT "Enter Device:CRT(C) or Plotter(P)",Dev$
80      PLOTTER IS "GRAPHICS"
90      GRAPHICS
100     IF Dev$="C" THEN GOTO 190
110     INPUT "Report(R) Mini(M) Oversize (O)",Size$
120     IF Size$="R" THEN OUTPUT 7,5;"IP1000,0,9000,6467"
130     IF Size$="M" THEN OUTPUT 7,5;"IP3000,4000,5000,5617"
140     IF Size$="O" THEN OUTPUT 7,5;"IP1000,0,15102,11400"
150     PLOTTER IS "9872A"
160     IF Size$="R" THEN INPUT "Viewgraph(V) or Paper(P)",Dp$
170     OUTPUT 7,5;"VS15"
180     IF Dp$="V" THEN OUTPUT 7,5;"VS5"
190     ASSIGN #1 TO File$
200     MAT READ #1;Plot
210     MAT SEARCH Plot(*,1),MAX;Max
211     MAT SEARCH Plot(*,1),MIN;Min
220     MAT SEARCH Plot(*,2),MAX;Mar
230     S=(1-Mar/Max)/2
240     IF Mar>Max THEN Max=Mar*1.2
241     Max=Max-Min
250     GCLEAR
260     INPUT "PLOT MENU: FULL, 1/2, 2/2, 1/4, 2/4, 3/4, 4/4",Menu$
270     IF Menu$="FULL" THEN 290
280     S=S/VAL(Menu$[3])
290     IF Menu$="FULL" THEN SCALE Min,Max,-S*Max,(1-S)*Max
300     IF Menu$="1/2" THEN SCALE Min,Max/2,-S*Max/2,(1-S)*Max/2
310     IF Menu$="2/2" THEN SCALE Max/2,Max,-S*Max/2,(1-S)*Max/2
320     IF Menu$="1/4" THEN SCALE Min,Max/4,-S*Max/4,(1-S)*Max/4
330     IF Menu$="2/4" THEN SCALE Max/4,Max/2,-S*Max/4,(1-S)*Max/4
340     IF Menu$="3/4" THEN SCALE Max/2,3*Max/4,-S*Max/4,(1-S)*Max/4
350     IF Menu$="4/4" THEN SCALE 3*Max/4,Max,-S*Max/4,(1-S)*Max/4
360     MAT PLOT Plot
370     PRINT PAGE
380     SCALE 0,123,0,100
390     MOVE 62,2
400     CSIZE 3.5,1
410     LORG 5
420     LABEL "Grid File = ";Name$
430                 W=1.5
440                 BEEP
450                 WAIT 300/W
460                 BEEP
470                 WAIT 150/W
480                 BEEP
490                 WAIT 150/W
500                 BEEP
510                 WAIT 300/W
520                 BEEP
530                 WAIT 600/W
540                 BEEP
550                 WAIT 300/W
560                 BEEP
570                 PEN 0
580     INPUT "Look at another section?(Y/N)",L$
590     IF L$="Y" THEN GOTO 250
600                 END
```

```
1       ! ******** PROGRAM "ELNUM" OF THE PREPPY PACKAGE ********
10      OPTION BASE 1
20      PRINTER IS 16
30      PLOTTER IS "GRAPHICS"
40      GRAPHICS
50      CALL Plots
60      END
70      SUB Plots
80      OPTION BASE 1
90      INPUT "Enter name of Supergrid file:",Sfile$
100     Sfile$=Sfile$&":C12"
110     ASSIGN #1 TO Sfile$
120     DIM Elem(800,11)
130     READ #1;Elemax
140     REDIM Elem(Elemax,11)
150     MAT READ #1;Elem
160     MAT SEARCH Elem(*,7),MAX;Zmax
170     MAT SEARCH Elem(*,8),MAX;Rmax
180     PRINT "Prepare platen for Large-Size paper, then CONT"
190     PAUSE
200     INPUT "Plot Menu: Full, 1/2, 2/2, 1/4, 2/4, 3/4, 4/4",Menu$
210     IF (Zmax>Rmax) AND (Menu$="FULL") THEN CALL Full(Zmax,A,B,D)
220     IF (Zmax>Rmax) AND (Menu$="1/2") THEN CALL One_half(Zmax,A,B,D)
230     IF (Zmax>Rmax) AND (Menu$="2/2") THEN CALL Two_half(Zmax,A,B,D)
240     IF (Zmax>Rmax) AND (Menu$="1/4") THEN CALL One_fourth(Zmax,A,B,D)
250     IF (Zmax>Rmax) AND (Menu$="2/4") THEN CALL Two_fourth(Zmax,A,B,D)
260     IF (Zmax>Rmax) AND (Menu$="3/4") THEN CALL Three_fourth(Zmax,A,B,D)
270     IF (Zmax>Rmax) AND (Menu$="4/4") THEN CALL Four_fourth(Zmax,A,B,D)
280     IF .99*Zmax<=Rmax THEN CALL Rscale(Rmax,A,B,D)
290     SCALE A,B,0,D
300             OUTPUT 7,5;"IP1000,0,15102,11400"
310             PLOTTER IS "9872A"
320             SCALE A,B,0,D
330             FOR I=1 TO Elemax
340             MOVE Elem(I,3)+E,Elem(I,4)+E
350             PEN 1
360             DRAW Elem(I,5)+E,Elem(I,6)-E
370             DEG
380             DRAW Elem(I,7)-E,Elem(I,8)-E
390             DRAW Elem(I,9)-E,Elem(I,10)+E
400             DRAW Elem(I,3)+E,Elem(I,4)+E
410             Ldir=0
420             LORG 5
430             IF Elem(I,6)-Elem(I,4)>Elem(I,7)-Elem(I,5) THEN Ldir=90
440             LDIR Ldir
450             CSIZE 1
460             Jim=(Elem(I,6)-Elem(I,4))/(B-A)*123
470             Jer=(Elem(I,7)-Elem(I,5))/D*100
480             IF (Ldir=0) AND ((Jim<.85) OR (Jer<2.3)) THEN 540
490             IF (Ldir=90) AND ((Jer<.85) OR (Jim<2.3)) THEN 540
500             Yax=(MAX(Elem(I,6),Elem(I,8))+MIN(Elem(I,4),Elem(I,10)))/2
510             Xax=(MAX(Elem(I,7),Elem(I,9))+MIN(Elem(I,3),Elem(I,5)))/2
520             MOVE Xax,Yax
530             LABEL I
540             NEXT I
590             SUBEXIT
600             SUBEND
610         SUB Full(Zmax,A,B,D)
620         A=0
630         B=Zmax
640         D=Zmax/1.23
650         SUBEXIT
660         SUBEND
670           SUB One_half(Zmax,A,B,D)
680           A=0
690           B=Zmax/2
```

```
700           D=Zmax/1.23/2
710           SUBEXIT
720           SUBEND
730         SUB Two_half(Zmax,A,B,D)
740         A=Zmax/2
750         B=Zmax
760         D=Zmax/1.23/2
770         SUBEXIT
780         SUBEND
790           SUB One_fourth(Zmax,A,B,D)
800           A=0
810           B=Zmax/4
820           D=Zmax/1.23/4
830           SUBEXIT
840           SUBEND
850         SUB Two_fourth(Zmax,A,B,D)
860         A=Zmax/4
870         B=Zmax/2
880         D=Zmax/1.23/4
890         SUBEXIT
900         SUBEND
910           SUB Three_fourth(Zmax,A,B,D)
920           A=Zmax/2
930           B=3*Zmax/4
940           D=Zmax/1.23/4
950           SUBEXIT
960           SUBEND
970         SUB Four_fourth(Zmax,A,B,D)
980         A=3*Zmax/4
990         B=Zmax
1000        D=Zmax/1.23/4
1010        SUBEXIT
1020        SUBEND
1030          SUB Rscale(Rmax,A,B,D)
1040          A=0
1050          B=Rmax*1.23*1.2
1060          D=Rmax*1.2
1070          SUBEXIT
1080          SUBEND
```

DISTRIBUTION LIST

No. Of Copies	Organization
12	Administrator Defense Technical Info Center ATTN: DTIC-DDA Cameron Station Alexandria, VA 22304-6145
1	Director Defense Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, VA 22209
1	Commander USA Materiel Command ATTN: AMCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333-0001
1	Commander USA Materiel Command ATTN: DRCLDC 5001 Eisenhower Avenue Alexandria, VA 22333
5	Commander USA AMCCOM, ARDC ATTN: SMCAR-SC SMCAR-LC SMCAR-SE SMCAR-SA SMCAR-AC Dover, NJ 07801
5	Commander USA AMCCOM, ARDC ATTN:SMCAR-SCS, D. Brandt SMCAR-SCS-E, J. Blumer SMCAR-SCF, G. DelCoco SMCAR-SCS, S. Jacobson SMCAR-SCF, K. Pfleger Dover, NJ 07801
3	Commander USA AMCCOM, ARDC ATTN: SMCAR-TDC, D. Gyorog SMCAR-TDA SMCAR-TDS Dover, NJ 07801
2	Commander USA AMCCOM, ARDC ATTN: SMCAR-TSS Dover, NJ 07801
6	Commander USA AMCCOM, ARDC ATTN: SMCAR-LCU, E. Barrieres SMCAR-LCU, R. Davitt SMCAR-LCU, D. Robertson SMCAR-LCU-M, J. Sikra SMCAR-LCU-M, M. Weinstock SMCAR-LCA-M, C. Larson Dover, NJ 07801
4	Commander USA AMCCOM, ARDC ATTN: SMCAR-LCA, B. Knutelski SMCAR-LCR-R, E.H. Moore, III SMCAR-LCS, J. Gregorits SMCAR-LCS, K. Rubin Dover, NJ 07801
7	Commander USA AMCCOM, ARDC ATTN: SMCAR-SCF, C.J. McGee SMCAR-SCA, S. Goldstein SMCAR-SCA, F.P. Puzychki SMCAR-SCA, E. Jeeter SMCAR-SCF, M.J. Schmitz SMCAR-SCF, L. Berman SMCAR-SCZ, P. Petrella Dover, NJ 07801
4	Commander USA AMCCOM, ARDC ATTN: SMCAR-LCA, S. Yim SMCAR-LCA, L. Rosendorf SMCAR-LCA, S.H. Chu SMCAR-LCW, R. Wrenn Dover, NJ 07801
1	Product Manager for 30mm Ammo ATTN: AMCPM-AAH-30mm Dover, NJ 07801

DISTRIBUTION LIST

No. Of Copies	Organization
7	Commander USA AMCCOM, ARDC ATTN: SMCAR-SCM, E. Bloore SMCAR-SCM, J. Mulherin SMCAR-SCS, B. Brodman SMCAR-SCS, T. Hung SMCAR-SCS, S. Jacobson SMCAR-SCA, W. Gadomski SMCAR-SCA, E. Malatesta Dover, NJ 07801
3	Director USA AMCCOM, ARDC Benet Weapons Laboratory ATTN: SMCAR-LCB-T SMCAR-LCB-TL SMCAR-LCB Watervliet, NY 12189
2	Director USA AMCCOM, ARDC Benet Weapons Laboratory ATTN: SMCAR-LCB-RA T. Simkins SMCAR-LCB-D, J. Zweig Watervliet, NY 12189
2	Commander USA AMCCOM, ARDC ATTN: SMCAR-SC, B. Shulman SMCAR-SC, Dover, NJ 07801
2	Commander USA AMCCOM, ARDC ATTN: SMCAR-SE Dover, NJ 07801
1	Commander USA AMCCOM, ARDC ATTN: Army Fuze Mgmt Proj Ofc SMCAR-FU Dover, NJ 07801
2	Commander USA AMCCOM, ARDC ATTN: Development Project Office For Selected Ammunitions SMCAR-DP Dover, NJ 07801
2	Commander USA AMCCOM, ARDC ATTN: Product Assurance Directorate SMCAR-QAR-RIB, D. Inhof Dover, NJ 07801
1	Commander USA AMCCOM, ARDC ATTN: SMCAR-NS Dover, NJ 07801
1	Commander USA AMCCOM, ARDC ATTN: L. Goldsmith Dover, NJ 07801
1	Commander USA Rock Island Arsenal ATTN: DRSMC-TSE-SW, R. Radkiewicz Rock Island, IL 61299
1	Commander USA AMMCCOM, ARDC ATTN: SMCAR-ESP-L, Tech Lib Rock Island, IL 61299
1	Commander USA Aviation Research and Development Command ATTN: AMSAV-E 4300 Goodfellow Blvd St. Louis, MO 63120
1	Director USA Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035
2	Director USA Research and Technology Laboratories (AVRADCOM) Ames Research Center Moffett Field, CA 94035

DISTRIBUTION LIST

No. Of Copies	Organization
1	Commander USA Communications Command ATTN: AMSEL-ED Fort Monmouth, NJ 07703
1	Commander USA Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703-5301
1	Commander USA Harry Diamond Labs ATTN: DELHD-TA-L 2800 Powder Mill Road Adelphi, MD 20783
1	Commander USA Missile Command Research, Development & Engineering Center ATTN: AMSMI-RD Redstone Arsenal, AL 35898
1	Commander USA Missile Command ATTN: AMSMI-R Redstone Arsenal, AL 35898
1	Commander USA Missile Command ATTN: AMSMI-RBL Redstone Arsenal, AL 35898
1	Commander USA Missile & Space Intelligence Center ATTN: AIAMS-YDL Redstone Arsenal, AL 35898-5500
1	Commander USA Natick Research and Development Command ATTN: DRXRE, H. Sieling Natick, MA 01762
2	Commander USA Tank Automotive Cmd ATTN: AMSTA-TSL Warren, MI 48090
2	Director USA TRADOC Systems Analysis Activity ATTN: ATAA-SL WSMR, NM 88002
2	Commander USA Research Office ATTN: R. Weigle E. Saibel P.O. Box 12211 Research Triangle Park, NC 27709
2	Commander USA Research Office P.O. Box 12211 ATTN: Technical Director Engineering Division Research Triangle Park, NC 27709
1	Commander USA Research Office ATTN: J. Chandra Research Triangle Park, NC 27709
2	Project Manager USA AMCCOM, ARDC Cannon Artillery Weapons System ATTN: AMCPM-CAWS Dover, NJ 07801
2	Project Manager Nuclear Munitions ATTN: AMCPM-NUC Dover, NJ 07801
2	Project Manager Tank Main Armament Systems ATTN: AMCPM-TMA Dover, NJ 07801

DISTRIBUTION LIST

No. Of Copies	Organization
2	Product Manager Division Air Defense Gun ATTN: AMCPM-ADG Dover, NJ 07801
2	Product Manager M110E2 Weapon System, DARCOM ATTN: AMCPM-M110E2 Rock Island, IL 61299
4	Director USA Mechanics and Materials Research Center ATTN: Director (3 cys) AMXMR-ATL (1 cy) Watertown, MA 02172
2	Commander USA Materials and Mechanics Research Center ATTN: J. Mescall Tech. Library Watertown, MA 02172
1	Commander Naval Air Systems Command ATTN: AIR-604 Washington, DC 20360
1	Commander Naval Ordnance Systems Command ATTN: ORD-9132 Washington, DC 20360
1	Commander Naval Sea Systems Command Washington, DC 20362
1	Commander Naval Ship Systems Command Washington, DC 20360
1	Commander Naval Ship Engineering Center Hyattsville, MD 20782
1	Superintendent Naval Postgraduate School ATTN: Dir of Lib Monterey, CA 93940
1	Commander Naval Air Development Center Johnsville Warminster, PA 18974
1	Commander Pacific Naval Missile Center Point Mugu, CA 95042
1	Commander Naval Research Laboratory Washington, DC 20375
2	Commander Naval Ship Research and Development Center Bethesda, MD 20084
4	Commander Naval Research Laboratory ATTN: W.J. Ferguson C. Sanday H. Pusey F. Rosenthal Washington, DC 20375
2	Commander Naval Surface Weapons Center ATTN: G-13, W.D. Ralph Dahlgren, VA 22448
1	Commander Naval Surface Weapons Center ATTN: Code X211, Lib Silver Spring, MD 20910
1	Commander Naval Surface Weapons Center ATTN: Code G-33, T.N. Tschirn Dahlgren, VA 22448
2	Commander Naval Weapons Center China Lake, CA 93555

DISTRIBUTION LIST

No. Of Copies	Organization
2	Commander Naval Weapons Center ATTN: Code 3835, R. Sewell Code 3431, Tech Lib China Lake, CA 93555
1	Commander Naval Ordnance Station Indian Head, MD 20640
2	Commandant USA Infantry School ATTN: ATSH-CD-CSO-OR Fort Benning, GA 31905
2	AFRPL ATTN: W. Andrepont T. Park Edwards AFB, CA 93523
1	AFOSR ATTN: W.J. Walker Bolling AFB, Washington, DC 20332
3	AFATL (DLA) Gun and Rocket Division ATTN: W. Dittrich, DLD D. Davis, DLDL Eglin, AFB, FL 32542
1	Air Force Armament Laboratory ATTN: AFTL/DLODL Eglin AFB, FL 32542-5000
1	Gun Test Branch AD 3246 Test W/TETFG Eglin AFB, FL 32542
2	AFWL/SUL Kirtland AFB, NM 87117
2	ASD Wright-Patterson AFB, OH 45433

No. Of Copies	Organization
1	ASD (ASAMCG) Tech Lib Wright-Patterson AFB, OH 45433
1	AFML (T. Nicholas) Wright-Patterson AFB, OH 45433
1	Director U.S. Bureau of Mines Twin Cities Research Center P.O. Box 1660 Minneapolis, MN 55111
3	Battelle Columbus Labs ATTN: L.E. Hulbert J.E. Backofen, Jr. 505 King Avenue Columbus, OH 43201
3	Battelle Pacific Northwest Laboratory ATTN: F.A. Simonen M. Smith P.O. Box 999 Richland, WA 99352
1	Lawrence Livermore National Laboratory ATTN: Dr. R.M. Christensen P.O. Box 808 Livermore, CA 94550
2	Sandia National Laboratory ATTN: Dr. C.W. Robinson Dr. Benidetti Applied Mechanics Department Engineering Mechanics Dept. Engineering Design Division Livermore, CA 94550
1	Director Lawrence Livermore Laboratory Livermore, CA 94550
1	Director Los Alamos Scientific Lab Los Alamos, NM 87544

DISTRIBUTION LIST

No. of Copies	Organization
2	National Aeronautics and Space Administration Langley Research Center Langley Station Hampton, VA 23365
1	Director National Aeronautics and Space Administration Manned Spacecraft Center ATTN: Library Houston, TX 77058
1	Director NASA - Ames Research Center ATTN: Tech Lib Moffett Field, CA 94035
1	Headquarters National Aeronautics and Space Administration Washington, DC 20546
	Cdr, CRDC, AMCCOM ATTN: SMCCR-RSP-A SMCCR-MU SMCCR-SPS-IL
1	HQDA DAMA-ART-M Washington, D.C. 20310
1	Commander U.S. Army Armament, Munitions and Chemical Command ATTN: SMCAR-ESP-L Rock Island, IL 61299
1	Commander U.S. Army Development and Employment Agency ATTN: MODE-TED-SAB Fort Lewis, WA 98433
10	Central Intelligence Agency Office of Central Reference Dissemination Branch Room GE-47 HQS Washington, D.C. 20502

No. of Copies	Organization
	Aberdeen Proving Ground
	Dir, USAMSAA ATTN: AMXSY-D AMXSY-MP, H. Cohen AMXSY-G, E. Christman AMXSY-OSD, H. Burke AMXSY-TN, D. Metz
	Cdr, USATECOM ATTN: AMSTE-CE AMSTE-TO-F
	Dir, USACSL, EA ATTN: SMCAR-CL SMCAR-CLB-PA, M. Miller SMCAR-CLB-PA, D. Olson SMCAR-CLB-PA, J. Huerta SMCAR-CLD SMCAR-CLY SMCAR-CLN SMCAR-CLN-D, L. Shaff SMCAR-CLN-D, F. Dagostin SMCAR-CLN-D, C. Hughes SMCAR-CLN-D, R. Simak SMCAR-CLN-D, M. Hayner
	Dir, USACSTA ATTN: Mr. Tag

USER EVALUATION SHEET/CHANGE OF ADDRESS

This Laboratory undertakes a continuing effort to improve the quality of the reports it publishes. Your comments/answers to the items/questions below will aid us in our efforts.

1. BRL Report Number_____ Date of Report_____

2. Date Report Received_____

3. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which the report will be used.)_____

4. How specifically, is the report being used? (Information source, design data, procedure, source of ideas, etc.)_____

5. Has the information in this report led to any quantitative savings as far as man-hours or dollars saved, operating costs avoided or efficiencies achieved, etc? If so, please elaborate._____

6. General Comments. What do you think should be changed to improve future reports? (Indicate changes to organization, technical content, format, etc.)

CURRENT ADDRESS

Name

Organization

Address

City, State, Zip

7. If indicating a Change of Address or Address Correction, please provide the New or Correct Address in Block 6 above and the Old or Incorrect address below.

OLD ADDRESS

Name

Organization

Address

City, State, Zip

(Remove this sheet along the perforation, fold as indicated, staple or tape closed, and mail.)

FOLD HERE

Director
U.S. Army Ballistic Research Laboratory
ATTN: SLCBR-DD-T
Aberdeen Proving Ground, MD 21005-5066

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, $300

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO 12062 WASHINGTON, DC
POSTAGE WILL BE PAID BY DEPARTMENT OF THE ARMY

Director
U.S. Army Ballistic Research Laboratory
ATTN: SLCBR-DD-T
Aberdeen Proving Ground, MD 21005-9989

FOLD HERE

END

11-86

DTIC